

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**

**FACULTAD DE INGENIERÍA**

**ESCUELA DE SISTEMAS**

**DISERTACIÓN DE GRADO**

**Análisis y diseño de un sistema de información para una empresa de transporte  
pesado, basado en la Metodología UML**

Johan Cedeño

**Quito, Octubre del 2015**

## **DEDICATORIA Y AGRADECIMIENTO**

Con mucho cariño y respeto:

A Dios, de quien todo depende. A mis padres y familia, quienes me formaron, educaron como una persona de bien y nunca perdieron la fe en mí. A mi esposa, quien día a día con su amor y ternura me alienta y respalda en este camino. A mis 2 hijos, que desde su inocencia me han dado las lecciones más valiosas y al mismo tiempo el impulso y la fortaleza de salir adelante. Al equipo de trabajo, Director y Revisores por aportar con su apertura, tiempo y paciencia. A todos quienes apostaron por mí y de los que recibí siempre su confianza, y a quienes dudaron ya que también me dieron más empuje para cumplir mis retos.

Contenido	
INTRODUCCIÓN .....	5
FUNDAMENTOS DEL PROBLEMA .....	7
ALCANCE .....	8
OBJETIVOS .....	8
General .....	8
Específicos .....	8
HIPÓTESIS .....	8
Capítulo I.....	9
El transporte pesado en el Ecuador .....	9
Capítulo II .....	13
Capítulo III .....	26
Capítulo IV .....	27
4. La Metodología UML (Unified Modelling Language) .....	27
4.1    Introducción .....	27
4.2 Historia .....	29
4.3 Concepto de UML.....	30
4.4 Modelos.....	33
4.4.1. Diagrama de Clases .....	35
4.5 Diagrama de casos de uso .....	36
4.6 Diagrama de Interacción .....	37
4.5. Diagrama de Actividades .....	40
4.7 DIAGRAMA DE ESTADO.....	42
4.8 DIAGRAMA DE OBJETOS .....	43
4.9 DIAGRAMA DE COMPONENTES .....	43
4.9 DIAGRAMAS DE DESPLIEGUE/DISTRIBUCION .....	45
Capítulo V .....	48
5. Análisis de Requerimientos del Sistema .....	48
5.1 Introducción .....	48

5.2 Sistema Actual .....	49
5.3 Sistema actual.....	52
5.4 Sistema propuesto .....	53
5.5 Requisitos no funcionales del sistema de software .....	63
5.6 Requisitos de rendimiento .....	63
5.7 Seguridad.....	64
5.8 Fiabilidad.....	64
5.9 Disponibilidad .....	64
5.10 Mantenibilidad .....	64
5.11 Portabilidad .....	64
5.12 Pseudorequerimientos o Restricciones .....	64
5.13 Modelos del Sistema .....	66
Capítulo VI.....	79
6. Diseño del Sistema .....	79
6.1 Introducción .....	79
6.2 Arquitectura del Sistema Actual .....	80
6.3 Arquitectura del Software Propuesto .....	80
6.3.1. Plataforma .....	81
6.3.2. Arquitectura Lógica.....	82
6.3.3. Arquitectura Física .....	86
6.4 Servicios de Subsistemas .....	87
Capítulo VI.....	91
7. Diagramas Estructurales UML.....	91
7.1 Introducción .....	91
7.2 Diagramas de clases .....	91
7.3 Diagrama de Componentes .....	92
7.4 Diagrama de paquetes .....	93
Capítulo VIII .....	103
8.1 Conclusiones .....	103
8.2 Recomendaciones.....	104
Bibliografía .....	105
ANEXOS.....	106

## Índice de Tablas

<b>Tabla 1 Rutas establecidas por JARCARGO.....</b>	<b>20</b>
<b>Tabla 2 Matriz de costos del proyecto.....</b>	<b>21</b>
<b>Tabla 3 Matriz de conclusiones.....</b>	<b>39</b>
<b>Tabla 4 Requisito funcional 1.....</b>	<b>47</b>
<b>Tabla 5 Requisito funcional 2.....</b>	<b>48</b>
<b>Tabla 6 Requisito funcional 3.....</b>	<b>49</b>
<b>Tabla 7 Requisito funcional 4.....</b>	<b>50</b>
<b>Tabla 8 Requisito funcional 5.....</b>	<b>51</b>
<b>Tabla 9 Requisito funcional 6.....</b>	<b>52</b>
<b>Tabla10 Requisito funcional 7.....</b>	<b>53</b>
<b>Tabla 11 Requisito funcional 8.....</b>	<b>54</b>
<b>Tabla 12 Requisito funcional 9.....</b>	<b>55</b>
<b>Tabla 13 Análisis de herramientas del sistema.....</b>	<b>69</b>
<b>Tabla 14 Matriz de plataformas.....</b>	<b>79</b>
<b>Tabla 15 Características del programa.....</b>	<b>80</b>
<b>Tabla 16 Matriz de herramientas del sistema.....</b>	<b>92</b>

## INTRODUCCIÓN

Con el antecedente de que no existe en las empresas ecuatorianas de transporte pesado un software personalizado similar al objeto de estudio presentado.

El propósito de este tema es formular una propuesta, para una empresa de Transporte Pesado Internacional en el proceso de globalización, sobre un sistema de información que maneje sus procesos de logística y mantenimiento de su flota. Esto se logrará, después de un análisis, estudio y estructuración de la misma, regido bajo la política de la empresa de dar un servicio de calidad para la completa satisfacción de sus clientes, e incorporar a la misma las certificaciones de procesos y calidad exigidos internacionalmente, más conocidos como ISO.

Para el análisis y diseño de esta problemática enfocada en el área de Sistemas se utilizará como apoyo una metodología para modelamiento de software que es el UML.

Para dar una solución se plantea el Lenguaje de Modelamiento Unificado (Unified Modeling Language) que es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software.

UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

## FUNDAMENTOS DEL PROBLEMA

La empresa objeto de aplicación fue creada en el año de 1988, se dedica al transporte de fluidos, y carga pesada en general, gracias a la iniciativa y dedicación de su presidente se ha ido colocando poco a poco como una de las principales empresas de transporte a escala nacional e internacional, ya que la misma realiza viajes a Colombia, Perú, Venezuela, entre otros.

El manejo de la empresa se lo hace bajo la supervisión directa de sus dueños, los mismos que controlan que se realicen correctamente los trabajos, y se dé un buen mantenimiento preventivo y correctivo a sus unidades.

Al tener un crecimiento muy grande en los últimos años tanto como empresa como en personal, se ha visto la necesidad de implementar un mayor control sobre el movimiento de sus bodegas, y crear sistemas de control de procesos de trabajo, y responsabilidad sobre los mismos, creando jerarquías y dando las responsabilidades necesarias a los mismos, entre las cuales tienen los mantenimientos preventivos y correctivos de la flota, servicios al cliente, recibimiento de órdenes de trabajo y despachos de los mismos, con los respectivos documentos y formularios requeridos por las leyes ecuatorianas y convenios internacionales.

Cabe destacar que todos estos procesos son manuales, por lo cual es preciso efectuar este análisis y diseño para presentar una propuesta a los directivos para el desarrollo de la misma.

En la parte de logística se quiere mantener un control en los gastos generados por los conductores al momento de realizar un servicio, y poderlo comparar con otros de igual complejidad para así estar al tanto de que tan reales son los reportes emitidos por los mismos (conductores); ejemplo de dichos gastos serían: combustible, hospedaje, alimentación, entre otros.

En fin, se busca optimizar el control de todo el personal de la empresa, lo que derivará en una reducción de costos que la beneficiará y permitirá brindar un mejor y más eficiente servicio.

## **ALCANCE**

Se realizará un análisis y diseño con la técnica UML para un sistema de automatización que maneje la logística y mantenimiento de la empresa llegando hasta las especificaciones de programación; para lo cual se efectuará una investigación teórica de lo que es y cómo se maneja la metodología en mención para el modelaje de software.

## **OBJETIVOS**

### **General**

Diseñar un sistema de información para una empresa de transporte pesado para su logística y mantenimiento, basado en la metodología UML

### **Específicos**

- Proponer mejoras en los procesos a partir del análisis.
- Brindar toda la documentación adecuada a la empresa, que facilite el desarrollo posterior del sistema integrado.
- Aprender y dominar la metodología UML.
- Resolver el problema de esta empresa con un análisis y diseño basados en la metodología UML.

## **HIPÓTESIS**

Esté análisis y diseño del sistema de automatización con su documentación podrá ser una herramienta suficiente para el posterior desarrollo e implementación del mismo, el cual podrá resolver los problemas que se han generado en esta empresa, apoyados en la información obtenida para esta propuesta.



## **Capítulo I.**

### **El transporte pesado en el Ecuador**

El sector del transporte pesado, que no tiene restricciones en las importaciones, está en crecimiento. El segmento de buses y camiones registró un 20% más de unidades vendidas en el 2014 frente al 2013. (AEADE, 2014)

Esa tendencia se mantiene en lo que va del presente año, sostienen las empresas del sector, producto de cuatro factores: el incremento de construcciones en el sector público y privado; la liquidez en el mercado, la apertura de créditos de la banca y el Plan Renova impulsado por el Régimen.

Por el lado estatal, los proyectos Coca Codo Sinclair, la Refinería del Pacífico, y carreteras han aportado al auge. En el sector privado, la construcción inmobiliaria también ha sido fundamental. En el país, 30 marcas de transporte pesado (camiones, tractores, buses, volquetes, etc.) se disputan el mercado: Kenworth, DAF, Mack, Volkswagen, Volvo, Renault, Jac, Hino, Mercedes Benz, son las principales de ellas.

El transporte de carga pesada tiene una fuerte penetración en el país, en lo que va del presente año se ha vendido un 20% más frente al año pasado. Las construcciones estatales y privadas son un pilar importante.

El sector presenta un crecimiento sostenido en los últimos tres años. Sin embargo, destaca que los precios de los camiones se han incrementado por el aumento del impuesto del 5% que estableció el Gobierno por la salida de divisas, valor que se trasladó al usuario.

A esto se suma el incremento de precios de las materias primas a escala internacional por los problemas económicos de Europa y EE.UU. (AEADE, 2014)

Actualmente, las unidades de transporte pesado oscilan en el mercado entre USD 130 000 y 200 000, según las marcas y modelos. Los equipos que tienen menor precio son los que se importan desde China.

Uno de los principales problemas que ha presentado la industria de transporte pesado es la informalidad. A partir del año 2012, quien quiera dedicarse a la transportación debe regularizarse y adquirir su permiso de operación a nivel de empresa y no de cooperativa. A esto se suma la implementación de la nueva ley de caminos.

Reglamentación que estuvo rezagada por 38 años pero que desde ahora se aplicará con rigor. Esta ley regula las dimensiones de los vehículos y el peso que pueden transportar según sus especificaciones técnicas, antes, los vehículos cargaban demasiado peso, lo que causa deterioro para el automotor y excesivo consumo de combustible. (FENATRAPE, 2014)

Según la FENATRAPE, entre el transporte formal y el informal circulan unos 115 mil vehículos de transporte pesado en el país, solo el 7% de vehículos obsoletos ha sido retirado de circulación en el país. La mayoría de ellos circula en la Costa.

Con respecto al mercado en general el quinto sector que más contribuyó al PIB en el 2014 fue el de transporte con el 0,39%. De hecho las cifras de venta de camiones también revelan un crecimiento del sector, las ventas de camiones subieron de 10 954 unidades en el 2013 a 11 085 en el 2014.

Las marcas más vendidas de estos vehículos fueron Chevrolet, Hino y JAC. (AEADE, 2014) Todos los sectores están relacionados con el transporte pesado, ya que ganaderos, industriales, constructores, embotelladores, comerciantes, supermercados, productores agrícolas, exportadores, farmacéuticos y demás necesitan transportar su materia prima, productos semi-elaborados o bienes terminados a los diferentes mercados de consumo.

“Los vehículos pesados son fundamentales para el traslado de materias primas y productos terminados que llegan por vía marítima, desde los puertos hasta las grandes ciudades. Además, son el sustento logístico del comercio con Colombia y Perú, ya que muchos productos salen e ingresan por tierra.” (FENATRAPE, 2014)

A nivel local, la utilización de camiones es fundamental en el movimiento interurbano de carga, desde los combustibles hasta las mudanzas, pasando por productos de primera necesidad y víveres que distribuyen de forma permanente en vehículos especialmente diseñados y equipados para cada caso.

Pero entre los pesados también se incluye a los vehículos de pasajeros. Los autobuses son el medio de transporte colectivo más usado, tanto en el servicio urbano como en el interprovincial.

Debido a las características físicas de los vehículos de transporte pesado y por el servicio que prestan, algunos municipios establecieron restricciones a su circulación mediante ordenanzas, para evitar el aumento de contaminación, tráfico y potenciales accidentes de tránsito.

Si bien los vehículos de transporte pesado tienen una vida útil muy extendida, en un determinado momento se hace necesaria la renovación de las unidades por varios motivos:

- Cumplir con las regulaciones municipales y policiales
- Precautelar la seguridad de los ocupantes y demás usuarios de las vías;
- Volver a la actividad más eficaz y económica.

Con mucha frecuencia, las empresas de transporte no utilizan vehículos propios, sino que contratan los servicios de transportadores a los que les pagan unos “fletes” por transportar la mercancía que les ha sido entregada por los generadores de carga.

Específicamente el mercado de transporte de carga pesada en la Ciudad de Quito es de 341 empresas que se dedican a esta actividad según la Superintendencia de Compañías.

En resumen:

- La industria de transporte pesado en Ecuador tuvo su auge entre los años 2007 y 2014 debido a las obras realizadas por el gobierno en lo que respecta a construcción vial, hidroeléctricas, entre otras.
- Actualmente en el mercado de la ciudad de Quito existen alrededor de 341 empresas dedicadas al transporte de carga pesada.
- Las salvaguardias implementadas por el gobierno en marzo del 2015 generan un riesgo para la industria de transporte pesado debido al incremento de artículos como

llantas y repuestos, sin embargo el verdadero impacto de esta medida se la puede realizar al finalizar la duración de la misma.

## Capítulo II

### Análisis General de la empresa Modelo

La empresa modelo a la que se realizará un análisis con la finalidad de implementar el sistema de información basado en metodología UML es JARCARGO CIA. LTDA, empresa ecuatoriana con más de 15 años en el mercado dedicada al transporte pesado, principalmente de:

- Productos líquidos limpios a granel
- Derivados de petróleo
- Productos químicos en general
- Cargas secas
- Refrigerados

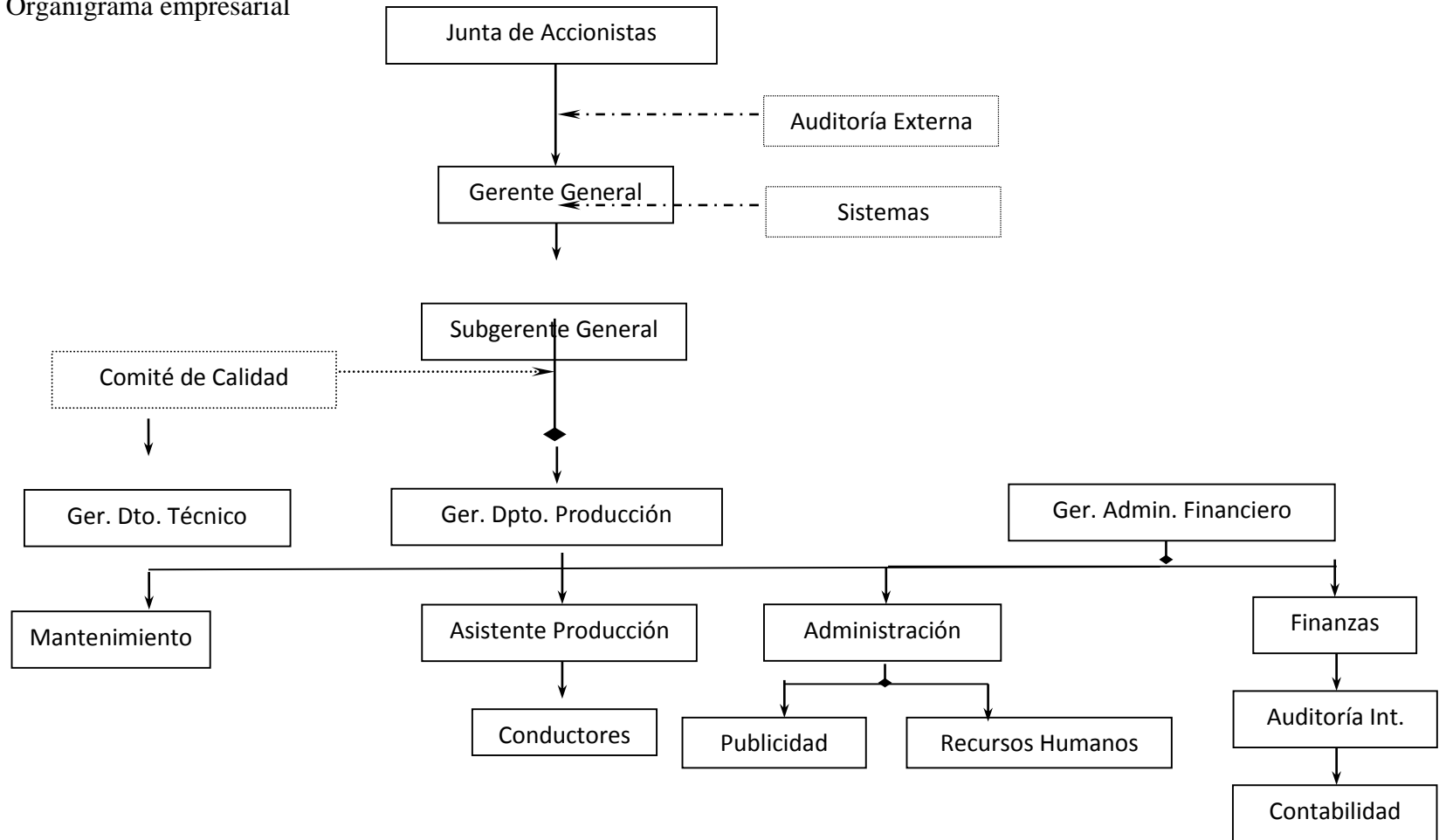
La empresa actualmente en su proceso de crecimiento ha obtenido las siguientes certificaciones que permiten su crecimiento dentro del mercado nacional e internacional:

- Norma ISO 9001
- ISO 14001
- OHSAS 18001
- Licencia Ambiental para el Transporte de Productos Químicos Especializados emitido por el MAE.
- Norma de control-seguridad BASC

Una de las principales problemáticas con las que cuenta la empresa es que no posee un sistema automatizado a través de un software para el manejo adecuado de la logística de transporte de la compañía, para lo cual el tema planteado solucionará el problema de la empresa y mejorará su sistema de transporte y por ende incrementará la eficiencia del servicio de la misma.

A continuación se presenta un organigrama estructurado con el que cuenta la empresa y las funciones de cada uno de los departamentos, ya que esta información ayudará al modelamiento del sistema en base a sus procesos y funciones.

Grafica 2.1 Organigrama empresarial



Fuente: JARCARGO Cía. Ltda.

Elaboración: Johan Cedeño. 2015

A continuación se presenta la descripción de cada departamento, enumerando las funciones que realiza cada miembro del mismo.

## **Gerencia**

- **Gerente General**

REPORTA A: Junta de Accionistas

SUPERVISA A: Subgerente General, Gerente Producción, Gerente Administrativo Financiero, Gerente Dpto. Técnico.

AREA: Gerencia

### **FUNCIONES Y RESPONSABILIDADES**

1. Tomar las decisiones adecuadas en lo referente a contratos y negocios que involucran a la empresa.
2. Revisar los reportes emitidos desde las gerencias de cada departamento y tomar las medidas necesarias.
3. Tener relación directa con los clientes para verificar la entera satisfacción de cada uno de ellos.
4. Buscar nuevas metas para el desarrollo de la empresa.
5. Supervisar la asistencia diaria, el correcto cumplimiento de las funciones y tareas encomendadas, y la buena presentación del personal de la empresa.
6. Promover la participación e interacción de los empleados.
7. Motivar en los empleados la adopción de la cultura de cero defectos.
8. Verificar el cumplimiento de los procedimientos implementados en la Empresa.
9. Establecer y cumplir procedimientos adecuados para que las fallas y errores cometidos no se repitan.
10. Renovar y revisar todos los procedimientos constantemente, hasta obtener un máximo de eficiencia y operatividad.
11. Cumplir y hacer cumplir el horario establecido en la empresa.



12. Cumplir con las responsabilidades encomendadas para el desarrollo, implementación y mantenimiento de la certificación de la Empresa Registrada bajo la norma ISO 9000.

- **Subgerente General**

REPORTA A: Gerente General

SUPERVISA A: Gerente Producción, Gerente Administrativo Financiero, Gerente Dpto. Técnico.

AREA: Gerencia

#### **FUNCIONES Y RESPONSABILIDADES**

1. Exigir revisar y analizar los reportes emitidos por las gerencias de cada departamento.
2. Buscar temas de capacitación al personal.
3. Emitir informes a la Gerencia General.
4. Participar con ideas y soluciones a los problemas encontrados en los reportes de cada departamento.
5. Liderar la solución de problemas y apoyar las nuevas ideas.
6. Supervisar la asistencia diaria, el correcto cumplimiento de las funciones y tareas encomendadas, y la buena presentación del personal de la Empresa.
7. Promover la participación e interacción de los empleados.
8. Motivar en los empleados la adopción de la cultura de cero defectos.
9. Verificar el cumplimiento de los procedimientos implementados en la Empresa.
10. Establecer y cumplir procedimientos adecuados para que las fallas y errores cometidos no se repitan.
11. Renovar y revisar todos los procedimientos constantemente, hasta obtener un máximo de eficiencia y operatividad.
12. Cumplir y hacer cumplir el horario establecido en la empresa.
13. Cumplir con las responsabilidades encomendadas para el desarrollo, implementación y mantenimiento de la certificación de Empresa Registrada bajo la norma ISO 9000.

## **Departamento de Producción o Logística**

### **• Gerente de Producción**

REPORTA A: Subgerente General

SUPERVISA A: Conductores en Operación (ruta, carga y descarga)

AREA: Departamento de Producción o Logística.

#### **FUNCIONES Y RESPONSABILIDADES:**

1. Servicio de ventas y post venta al cliente.
2. Programar y Coordinar viajes Nacionales e Internacionales.
3. Realizar órdenes de despacho de vehículos desde el patio.
4. Realizar documentación de transporte Internacional de Importación y Exportación.
5. Realizar seguimiento de documentación para cruce de frontera.
6. Tener al día la documentación legal para todos los vehículos.
7. Tener al día la documentación legal de la empresa para el servicio de transporte Nacional e Internacional.
8. Facturación a clientes.
9. Reporte mensual de producción.
10. Actualizar diariamente en el reporte de novedades los casilleros de Ruta, Ubicación, llegada al Patio, Producto, y Observaciones si existe alguna novedad.
11. Controlar eficazmente la producción y optimización de los procesos en el Departamento.
12. Supervisar la asistencia diaria, el correcto cumplimiento de las funciones y tareas encomendadas, y la buena presentación del personal del Departamento.
13. Promover la participación e interacción de los empleados.
14. Motivar en los empleados la adopción de la cultura de cero defectos.
15. Verificar el cumplimiento de los procedimientos implementados en el Departamento.
16. Establecer y cumplir procedimientos adecuados para que las fallas y errores cometidos no se repitan.
17. Renovar y revisar todos los procedimientos constantemente, hasta obtener un máximo de eficiencia y operatividad.
18. Controlar el orden y aseo del Departamento.

19. Cumplir y hacer cumplir el horario establecido en la empresa
20. Cumplir con las responsabilidades encomendadas para el desarrollo, implementación y mantenimiento de la certificación de Empresa Registrada bajo la norma ISO 9000.
21. Cumplir con todo encargo adicional que permita el buen funcionamiento de la empresa.

## **Departamento Administrativo Financiero**

- **Gerente del Dpto. Administrativo - Financiero**

REPORTA A: Gerencia y Subgerencia General

SUPERVISA A: Asistente del Dpto., Recepcionista.

AREA: Departamento Administrativo - Financiero.

### **FUNCIONES Y RESPONSABILIDADES:**

1. Relaciones bancarias, control diario de saldos bancarios.
2. Control de pagos.
3. Control de ingresos.
4. Flujos de caja.
5. Relación directa con gerentes de empresas, asunto económico, cobranzas.
6. Elaboración de presupuestos.
7. Elaboración de balances, firma de estados financieros.
8. Preparar información para SRI.
9. Preparar informes para gerencia general.
10. Recomendaciones para el departamento técnico, logística y sistemas de acuerdo a los ingresos y egresos.
11. Aprobación de roles de pago.
12. Contestar comunicaciones relacionadas.
13. Liquidación subcontratos.
14. Asuntos relacionados directamente con contabilidad.
15. Registro numérico de cartas portes y sobordos.
16. Controlar eficazmente la producción y optimización de los procesos en el Departamento.
17. Supervisar la asistencia diaria, el correcto cumplimiento de las funciones y tareas encomendadas, y la buena presentación del personal del Departamento.
18. Promover la participación e interacción de los empleados.
19. Motivar en los empleados la adopción de la cultura de cero defectos.
20. Verificar el cumplimiento de los procedimientos implementados en el Departamento.

21. Establecer y cumplir procedimientos adecuados para que las fallas y errores cometidos no se repitan.
22. Renovar y revisar todos los procedimientos constantemente, hasta obtener un máximo de eficiencia y operatividad.
23. Controlar el orden y aseo del Departamento.
24. Cumplir y hacer cumplir el horario establecido en la empresa.
25. Cumplir con las responsabilidades encomendadas para el desarrollo, implementación y mantenimiento de la certificación de Empresa Registrada bajo la norma ISO 9000.
26. Cumplir con todo encargo adicional que permita el buen funcionamiento de la empresa.

**Asistente del Dpto. Administrativo - Financiero**

REPORTA A: Gerente del Dpto. Administrativo - Financiero

AREA: Departamento Administrativo - Financiero.

**FUNCIONES Y RESPONSABILIDADES:**

1. Cobro clientes.
2. Pago proveedores.
3. Elaboración de roles de pago.
4. Realizar cuentas con los conductores.
5. Formularios para pagos impuestos SRI.
6. Formularios para pagos aportes IEES.
7. Digitación de egresos e ingresos programa.
8. Coordinar depósitos con los conductores para viajes.
9. Cumplir con exactitud todos los procedimientos implementados en el Departamento.
10. Mantener el orden y aseo del Departamento.
11. Cumplir con las responsabilidades encomendadas para el desarrollo, implementación y mantenimiento de la certificación de Empresa Registrada bajo la norma ISO 9000.
12. Cumplir con todo encargo adicional que permita el buen funcionamiento de la empresa.

## **Departamento Técnico**

- **Gerente del Departamento Técnico**

REPORTA A: Subgerente General  
 SUPERVISA A: Unidad en mantenimiento.  
 AREA: Departamento Técnico

### **FUNCIONES Y RESPONSABILIDADES:**

1. Garantizar el mantenimiento y buen funcionamiento de los cabezales y Tanques.
2. Controlar eficazmente la producción y optimización de los procesos en todas las áreas del Patio (mecánica, electricidad, enderezada y pintura, lavadora, bodega, suelda, vulcanizadora, compras, etc.).
3. Elaborar el reporte diario de unidades, especificando los trabajos a realizarse en el Taller durante el día dando un tiempo aproximado de reparación.
4. Supervisar la asistencia diaria, el correcto cumplimiento de las funciones y tareas encomendadas, y la buena presentación del personal del Departamento.
5. Promover la participación e interacción de los empleados.
6. Motivar en los empleados la adopción de la cultura de cero defectos.
7. Revisar la hoja de reporte de fallas de las Unidades, analizar y asignar el personal respectivo para el mantenimiento si es necesario.
8. Revisión y aprobación de cotizaciones y órdenes de compra.
9. Planificar y distribuir los trabajos de mantenimiento del personal técnico.
10. Verificar el cumplimiento de los procedimientos implementados en el Patio.
11. Control de vacaciones del personal técnico.
12. Tramitar horas extras del personal del patio.
13. Involucrarse totalmente en los problemas y reparaciones importantes para su inmediata y correcta solución.
14. Establecer y cumplir procedimientos adecuados para que las fallas y errores cometidos no se repitan.

15. Renovar y revisar todos los procedimientos constantemente, hasta obtener un máximo de eficiencia y operatividad.
16. Controlar el orden y aseo de todas las áreas del Patio.
17. Verificar la calidad de los trabajos de mantenimiento realizados.
18. Cumplir turno de control de fin de semana alternando con los Jefes de Adquisiciones y Patio.
19. Cumplir y hacer cumplir el horario establecido en la empresa.
20. Entregar los primeros días de cada mes el informe y reporte de gastos de mantenimiento real por unidad y total del área de mantenimiento, reportes de novedades de personal y conductores.
21. Negociar y aprobar el costo de los trabajos con los contratistas de la empresa (soldador, electricista y pintor).
22. Cumplir con las responsabilidades encomendadas para el desarrollo, implementación y mantenimiento de la norma ISO 9000.
23. Cumplir con todo encargo adicional que permita el buen funcionamiento de la empresa.



Tabla 1. Rutas establecidas por JARCARGO

RUTAS ESTABLECIDAS POR PAIS					
ECUADOR	COLOMBIA	PERU	BOLIVIA	VENEZUELA	CHILE
QUITO	IPIALES	TUMBEZ	LA PAZ	VALENCIA	SANTIAGO
GUAYAQUIL	PASTO	TALARA			
SANTO DOMINGO	CALI	SULLANA			
EL EMPALME	MEDELLIN	TRUJILLO			
PUERTO CAYO	BUCARAMANGA	LIMA			
MACHALA	MANIZALES	TACNA			
BABAHOYO	ARMENIA				
LATACUNGA	BOGOTA				
AMBATO	CARTAGENA				
RIOBAMBA	SANTA MARTHA				
ESMERALDAS	BARRANCABERMEJA				
LAGO AGRIO	BARRANQUILLA				
EL CHACO	IBAGUE				
BAEZA	NEIVA				
TENA	PALMIRA				
PUYO	POPAYAN				
EL COCA	TUNJA				
SACHA	VALLEDUPAR				
CUYABENO					
SHUSHUFINDI					
CUENCA					
CAÑAR					
SANTA ELENA					
ALASI					
PORTOVIEJO					
MACAS					
LOJA					
ZAMORA					
AZOGUES					

Fuente: [www.jarcargo.com](http://www.jarcargo.com) elaboración: Johan Cedeño. 2015

En resumen:

Aunque el organigrama de la empresa se encuentra bien estructurado, es recomendable implementar áreas comerciales, como es el caso de un departamento de Marketing o Ventas con la finalidad de incrementar su posicionamiento como sus utilidades.

## Capítulo III

### Costos

En el presente capítulo se detalla una matriz de costos para la implementación del proyecto por parte de la empresa.

Tabla 2 Matriz de costos del proyecto

MATRIZ DE COSTOS DEL PROYECTO	
Costos de desarrollo del software	
	Inversión requerida
Programador	\$ 4.000,00
Supervisor	\$ 1.500,00
Costos de equipo para el desarrollo	
2 servidores	\$ 2.500,00
Internet	\$ 50,00
Otros costos	
Imprevistos	\$ 300,00
<b>TOTAL DE LA INVERSION</b>	<b>\$ 8.350,00</b>

Elaboración: Johan Cedeño. 2015

Como se puede apreciar en la tabla de costos del proyecto la inversión total para la implementación del sistema es de \$8350 que involucra el desarrollo e implementación del software así como de los equipos necesarios y los gastos adicionales que intervienen en el mismo.

El tiempo de ejecución del proyecto va a ser de 6 meses en los cuales se trabajará con un programador y un supervisor; además de dos servidores y se utilizarán las instalaciones de la empresa JARCARGO para la elaboración del trabajo.

## **Capítulo IV**

### **4. La Metodología UML (Unified Modelling Language)**

#### **4.1 Introducción**

En el ciclo de vida del software dentro de la planificación informática de una organización es importante considerar un lenguaje estándar y unificado que permita modelar de forma estructurada y documentar la calidad de los proyectos y aplicaciones de software de manera eficiente.

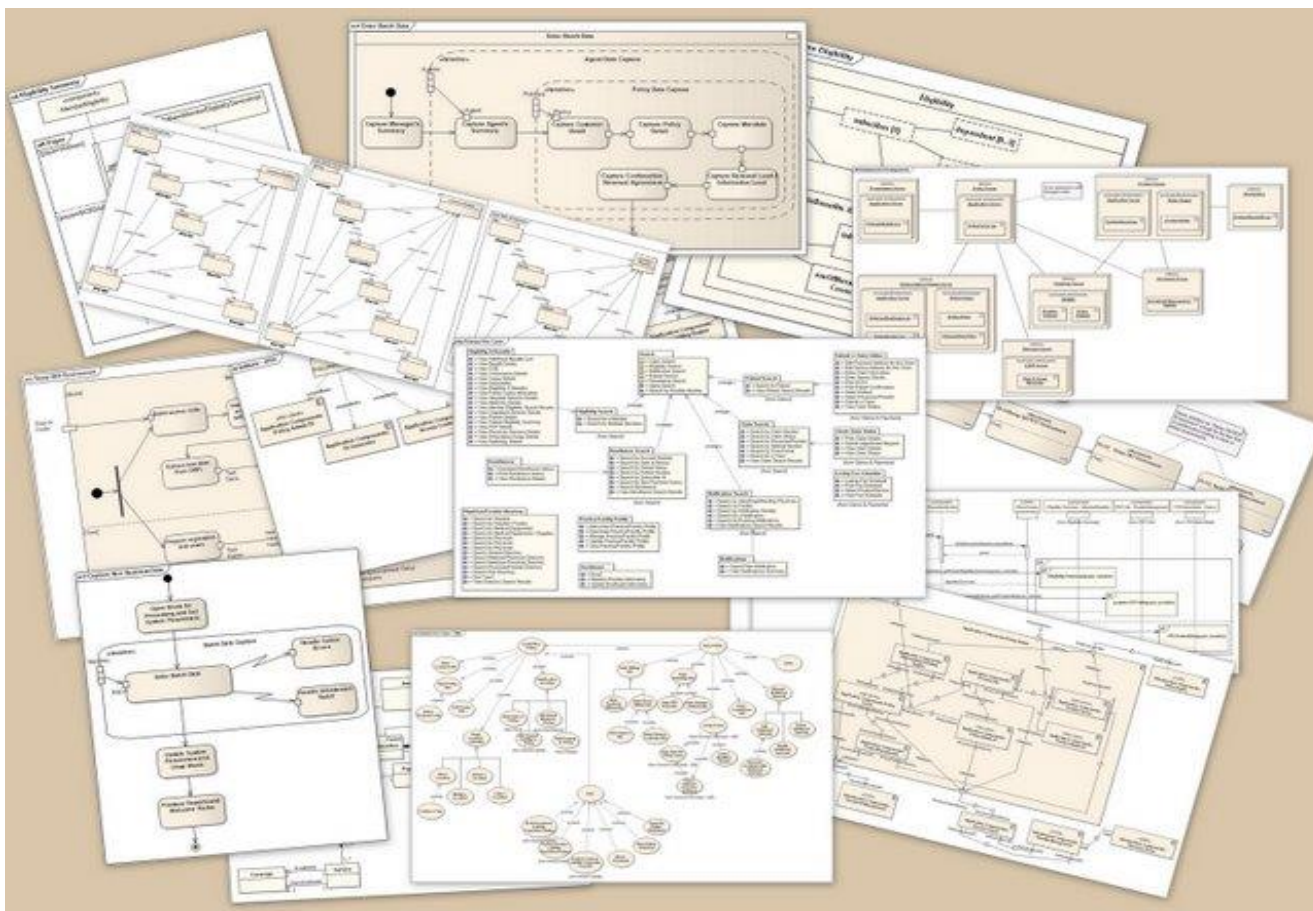
Un lenguaje común, visualmente fácil de comprender y que abarque los diferentes aspectos de las aplicaciones y sistemas traerá una optimización de recursos en el proceso del desarrollo, una proyección en el tiempo en base a las necesidades de la organización, un mejor despliegue de servicios y una minimización de reprocesos y problemas por corrección de errores.

Al ser un lenguaje común en el ámbito informático permite disponer de fuentes de información y una fluidez en el mercado del software para dar soporte, compartir en la comunidad y madurar los procesos en la inherente madurez de los sistemas.

Al ser visualmente fácil para modelar los sistemas informáticos se convierte en comprensible y dispone de mayor amplitud de planificación, seguimiento y corrección del sistema. Su especificación y documentación se mantiene en un lenguaje amigable.

Este lenguaje además es muy importante ya que comprende aspectos conceptuales tales como procesos de negocios y funciones del sistema.

Es una herramienta sólida en la etapa del levantamiento de requerimiento y diseño funcional, lo que ayuda a evitar malos entendidos entre las partes interesadas e involucrados del proyecto.

Gráfica 4.1. Lenguaje unificado de modelado<sup>1</sup>

Fuente: [http://es.wikipedia.org/wiki/Lenguaje\\_unificado\\_de\\_modelado](http://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado). 2015

“Este lenguaje de modelado pretende unificar la experiencia pasada sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar”.  
(Wikipedia, 2013)

<sup>1</sup> [http://es.wikipedia.org/wiki/Lenguaje\\_unificado\\_de\\_modelado](http://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado)

## 4.2 Historia

El lenguaje UML fue el resultado inicial de la fusión de las metodologías de Grady Booch y de James Rumbaugh en el año de 1994 en el intento de generar un lenguaje para especificar y documentar un sistema de software.

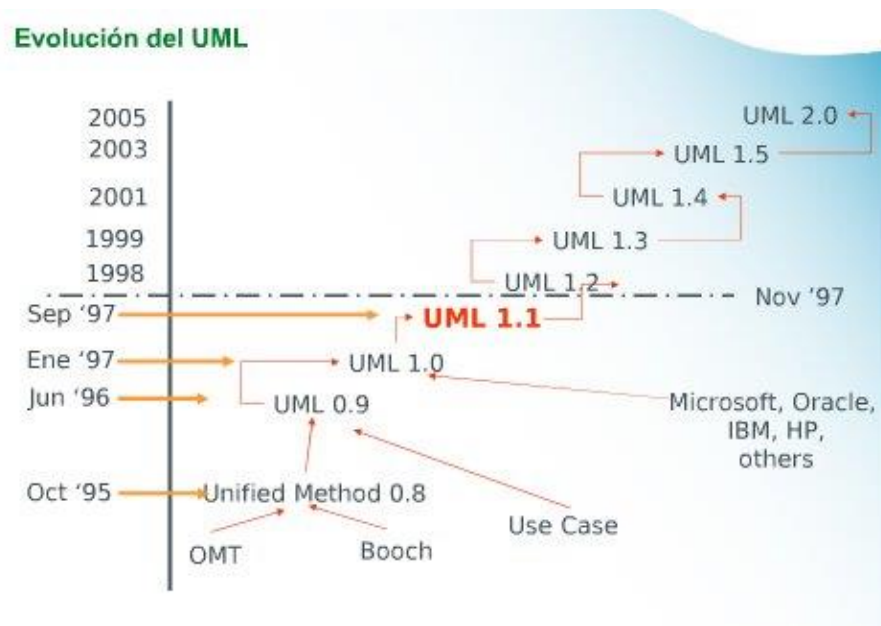
El objetivo de ambos era unificar la metodología de Booch que abarcaba la descripción de conjuntos de objetos y sus relaciones con la técnica de modelado orientada a objetos de James Rumbaugh (OMT: Object - Modelling Technique).

Adicionalmente y con la colaboración de Ivar Jacobson se implementó la metodología de casos de uso (OOSE: Object- Oriented Software Engineering).

Mientras las metodologías de Booch y Rumbaugh se centran en el modelado de los objetos que componen el sistema, su relación y colaboración, la metodología de Jacobson es más centrada a usuario, ya que todo en su método se deriva de los escenarios de uso.

En 1997 UML 1.1 fue aprobada por la OMG convirtiéndose en mayor especificación para el análisis y el diseño orientado a objetos.

Gráfica 4.2. Evolución del UML<sup>2</sup>

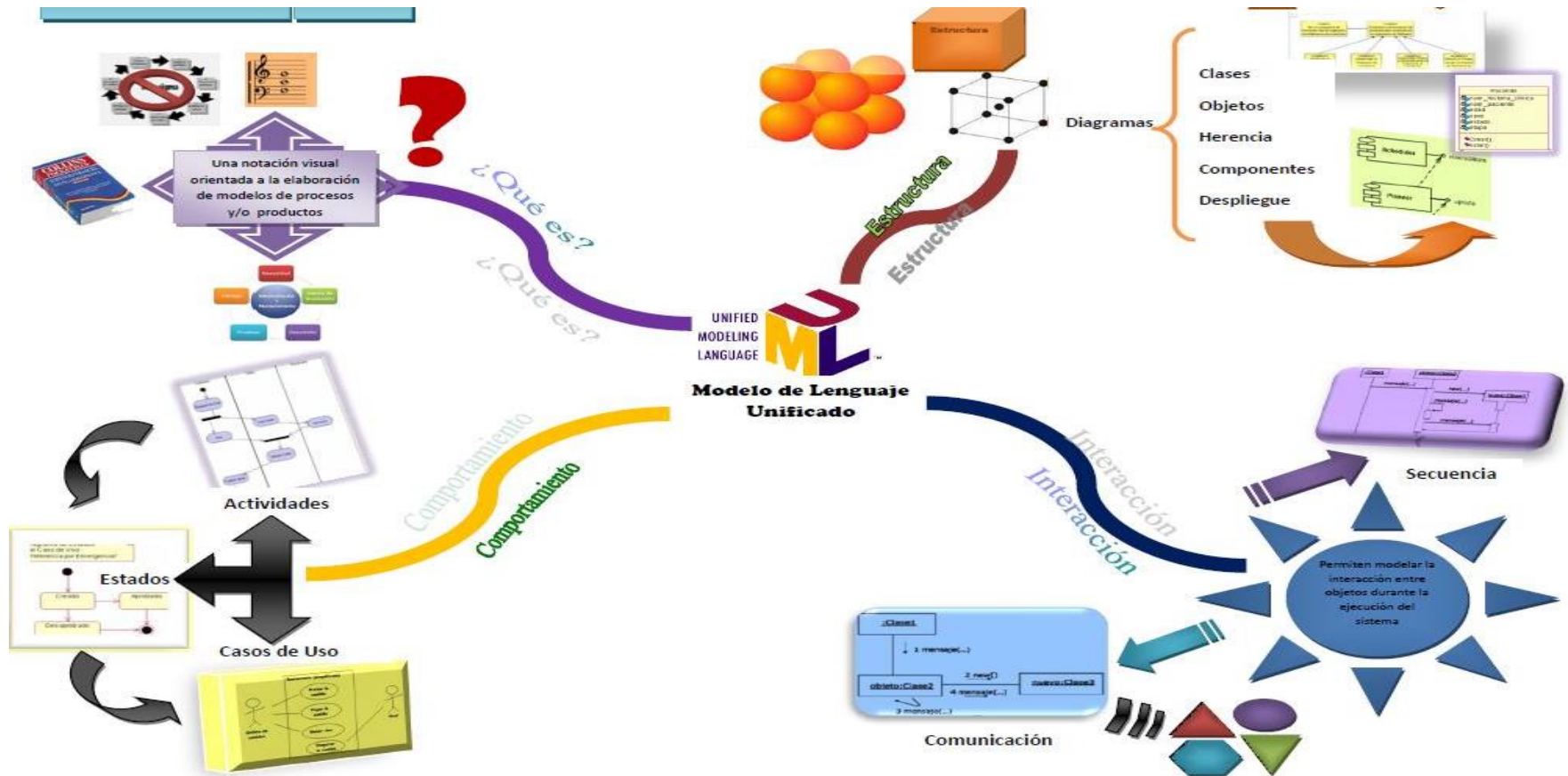


Fuente: <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>. 2015

<sup>2</sup> <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>

### **4.3 Concepto de UML**

“El Unified Modelling Language (UML) es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Se usa para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir” (Perez, 2012)

Gráfica 4.3. Modelo UML <sup>3</sup>

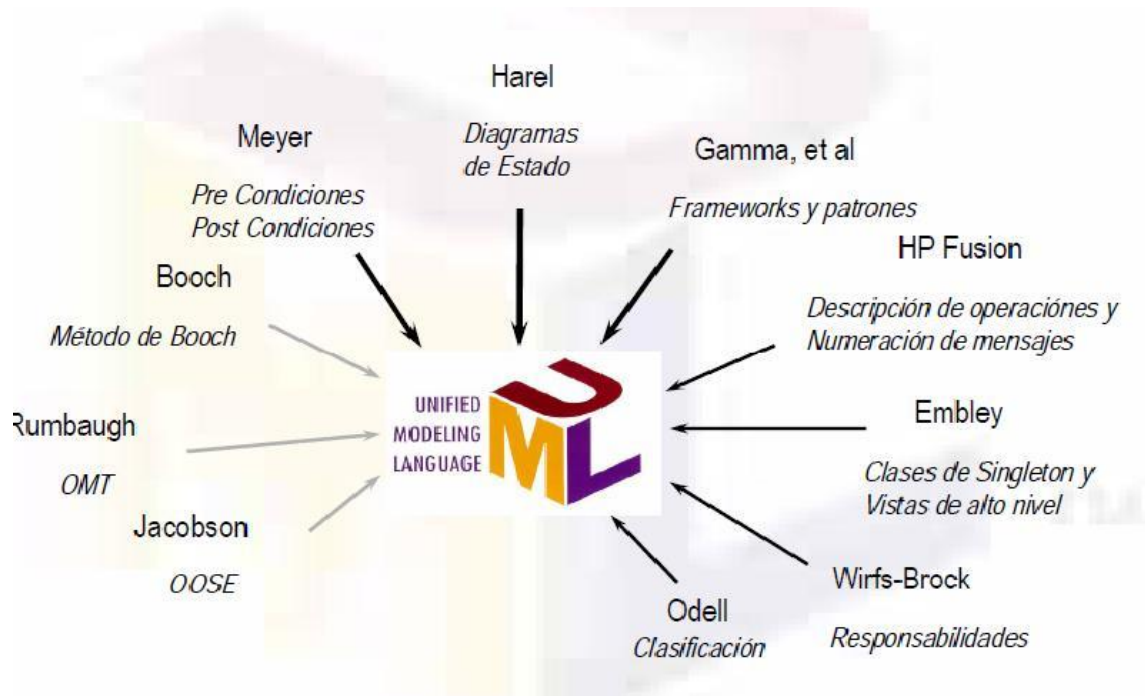
Fuente: [http://softwareverde.blogspot.com/2012\\_09\\_01\\_archive.html](http://softwareverde.blogspot.com/2012_09_01_archive.html). 2015

<sup>3</sup> [http://softwareverde.blogspot.com/2012\\_09\\_01\\_archive.html](http://softwareverde.blogspot.com/2012_09_01_archive.html)

UML se puede visualizar en tres aspectos o vistas principalmente donde cada una describe una colección de objetos discretos que interactúan. Una vista estructural lógico que describe los elementos del sistema y sus relaciones (diagramas de clases y los diagramas de casos de uso).

Una dinámica que describe las interacciones entre los objetos con el tiempo (diagramas de interacción, las máquinas de estado y los diagramas de actividades). Una vista de implementación en la que se describe la estructura física del sistema en cuanto a los componentes de que consta y su ubicación (diagramas de componentes y de despliegue). (Perez, 2012)

Gráfica 4.4 Modelo de lenguaje UML



Fuente: <http://2.bp.blogspot.com /uml.jpg>. 2015

<sup>4</sup><http://2.bp.blogspot.com /uml.jpg>



Es importante recalcar que UML no es un lenguaje de programación sin embargo en las herramientas de desarrollo se encuentran normalmente generadores de código de UML para una gran variedad de lenguaje de programación.

Además se dispone de reingeniería inversa en la que se generan los diagramas para UML a partir del código de programación.

Según Enrique Hernández Orallo los objetivos de UML se pueden sintetizar en las siguientes funciones:

- **Visualizar:** UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- **Especificar:** UML permite especificar cuáles son las características de un sistema antes de su construcción.
- **Construir:** A partir de los modelos especificados se pueden construir los sistemas diseñados.
- **Documentar:** Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión. (Orallo, 2001)

Aunque el UML define un lenguaje preciso, no es impedimento para el desarrollo de conceptos de modelaje más avanzados, más aun sirve de base sobre los que se puede construir especificaciones sin perder su esencia inicial.

El objetivo que tiene UML es crear una documentación común, que permita a los desarrolladores con conocimientos de UML entender los aspectos del sistema, independientemente del lenguaje utilizado para el desarrollo.

#### 4.4 Modelos

Por definición un modelo es una representación de un objeto, sistema o idea, de forma diferente al de la entidad misma. El propósito de los modelos es ayudarnos a explicar, entender o mejorar un sistema. Un modelo de un objeto puede ser una réplica exacta de éste o una abstracción de las propiedades dominantes del objeto. (Bermón, 2014)

En un sistema de software un modelo abstrae los elementos, relaciones y/o interacciones desde una perspectiva específica como la interacción del usuario con el sistema, las entidades de información o los componentes físicos del sistema. En UML se dispone de los siguientes elementos:

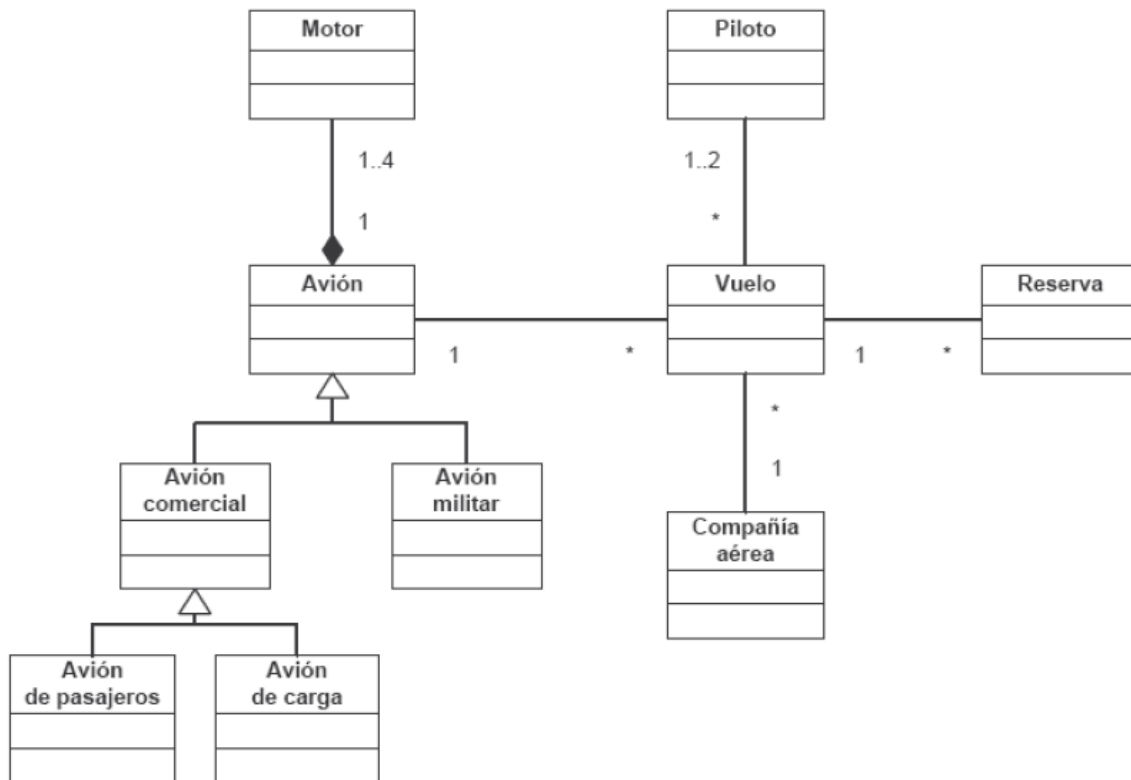
- Diagrama de Casos de Uso
- Diagrama de Clases
- Diagramas de Comportamiento
  - Diagrama de Estados
  - Diagrama de Actividades
- Diagramas de Iteración
  - Diagrama de Secuencia
  - Diagrama de Colaboración
- Diagramas de Implementación
  - Diagrama de Componentes
  - Diagrama de Despliegue

Se puede clasificar el modelo de una forma estática o dinámica, de esta manera tenemos:

- Modelo Estático (estructural)
  - Diagrama de Despliegue
  - Diagrama de componentes.
  - Diagrama de clases
  - Diagrama de objetos
- Modelo dinámico (comportamientos)
  - Diagrama de estados
  - Diagrama de actividades
  - Diagrama de secuencia
  - Diagrama de colaboración
  - Diagrama de casos de uso

#### 4.4.1. Diagrama de Clases

Gráfica 4.5. Diagrama de clases<sup>5</sup>



Fuente: <http://www.italisite.info/>. 2015

Un diagrama de clases es una representación gráfica estática que describe la estructura de un sistema sus clases, atributos, relaciones entre ellos. Estos diagramas son utilizados durante el proceso de análisis y diseño del sistema de software.

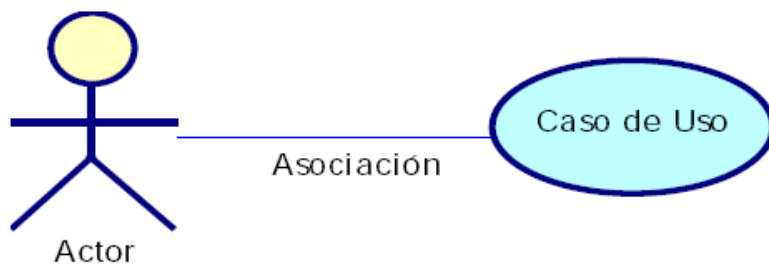
Ese diagrama describe las entidades o conceptos a manera de cajas negras pero que internamente describen atributos y operaciones específicas.

<sup>5</sup> Fuente: <http://www.italisite.info/>. 2015

#### 4.5 Diagrama de casos de uso

Un Diagrama de Casos de Uso es un diagrama de comportamiento UML mejorado en el que se muestra la relación entre los actores y los casos de uso del sistema. Representa la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa.

Gráfica 4.5. Diagrama de casos de uso<sup>6</sup>



Fuente: [www.google.com/diagramadecasodeuso/imagenes](http://www.google.com/diagramadecasodeuso/imagenes). 2015

- **Elementos**

Los elementos que pueden aparecer en un Diagrama de Casos de Uso son: actores, casos de uso y relaciones entre casos de uso.

- **Actor**

Un actor es una entidad externa al sistema que realiza algún tipo de interacción con el mismo. Es un rol que un usuario juega con respecto al sistema. Se representa mediante una figura humana dibujada con líneas. Esta sirve tanto para actores que son personas como para otro tipo de actores.

- **Casos de Uso**

Un caso de uso es una descripción de la secuencia de interacciones, operaciones o acciones específicas que se producen entre un actor y el sistema. Expresa una unidad coherente de

---

<sup>6</sup> [www.google.com/diagramadecasodeuso/imagenes](http://www.google.com/diagramadecasodeuso/imagenes)

funcionalidad, y se representa mediante una elipse con el nombre del caso de uso en su interior.

- **Relaciones entre Casos de Uso**

Las tres relaciones principales entre los casos de uso:

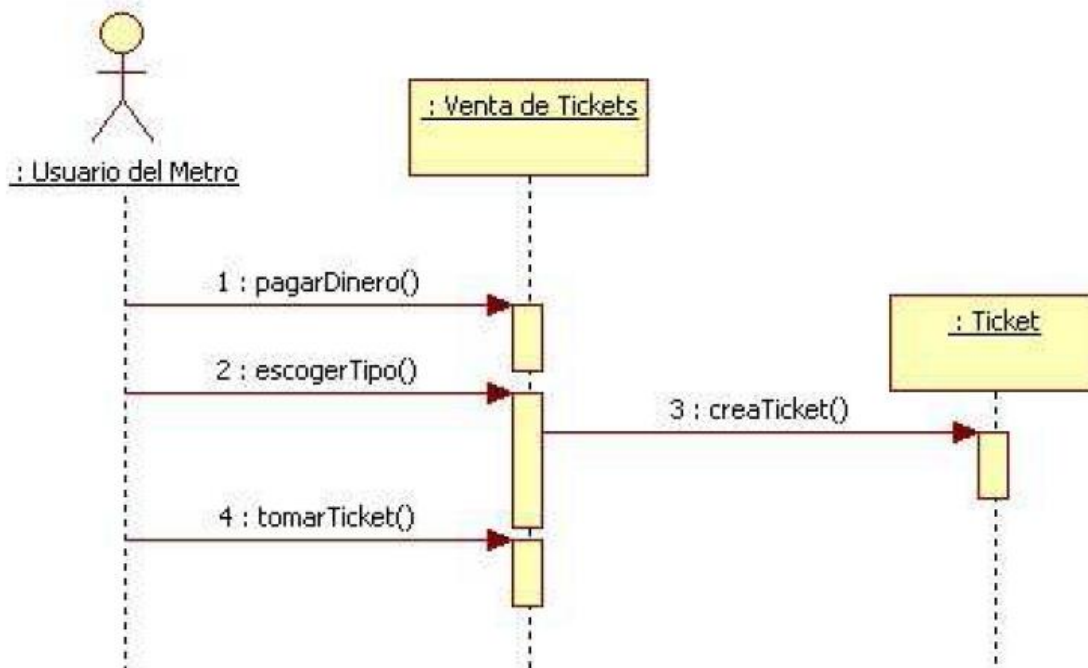
- **Extiende:** Cuando un caso de uso especializa a otro extendiendo su funcionalidad.
- **Usa:** Cuando un caso de uso utiliza a otro.
- **Generaliza:** La actividad de identificar elementos en común entre conceptos y definir las relaciones.

#### **4.6 Diagrama de Interacción**

Los diagramas de interacción son una representación gráfica que muestra un patrón de interacción entre objetos. Existen dos tipos de diagrama de interacción basados en la misma información: Diagramas de Secuencia y Diagramas de Colaboración.

## Diagrama de Secuencia

Gráfica 4.6. Diagrama de secuencia<sup>7</sup>



Fuente: [www.google.com/diagramadesecuencias/imagenes](http://www.google.com/diagramadesecuencias/imagenes) 2015

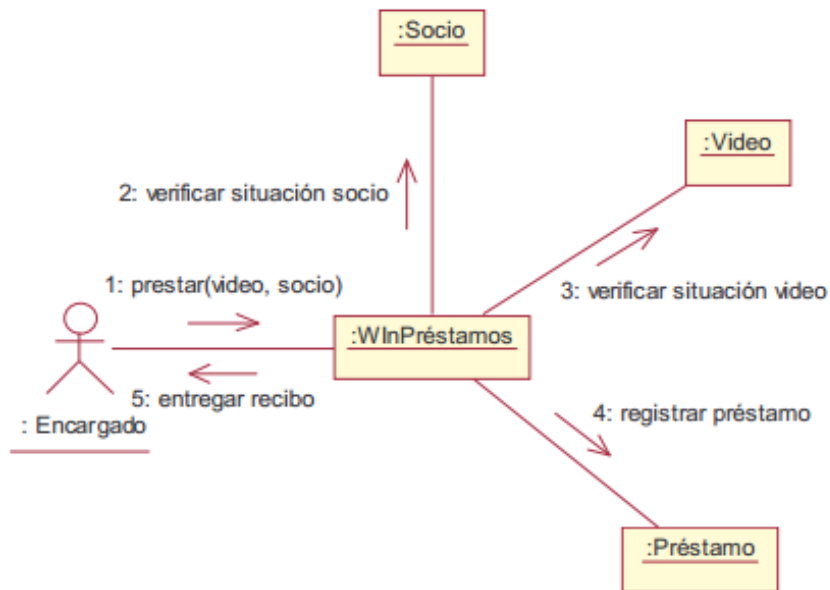
Un diagrama de Secuencia muestra una interacción ordenada en el tiempo de eventos. Muestra los objetos participantes en la interacción y los mensajes que intercambian ordenados según su secuencia en el tiempo.

El eje vertical representa el tiempo, y en el eje horizontal se colocan los objetos y actores participantes en la interacción, sin un orden prefijado. Cada objeto o actor tiene una línea vertical, y los mensajes se representan mediante flechas entre los distintos objetos. El tiempo fluye de arriba abajo. Se pueden colocar etiquetas (como restricciones de tiempo, descripciones de acciones, etc.) bien en el margen izquierdo o bien junto a las transiciones o activaciones a las que se refieren.

<sup>7</sup> [www.google.com/diagramadesecuencias/imagenes](http://www.google.com/diagramadesecuencias/imagenes)

### Diagrama de Colaboración

Gráfica 4.7. Diagrama de colaboración<sup>8</sup>



Fuente: [www.google.com/diagramadecolaboración/imágenes](http://www.google.com/diagramadecolaboración/imágenes). 2015

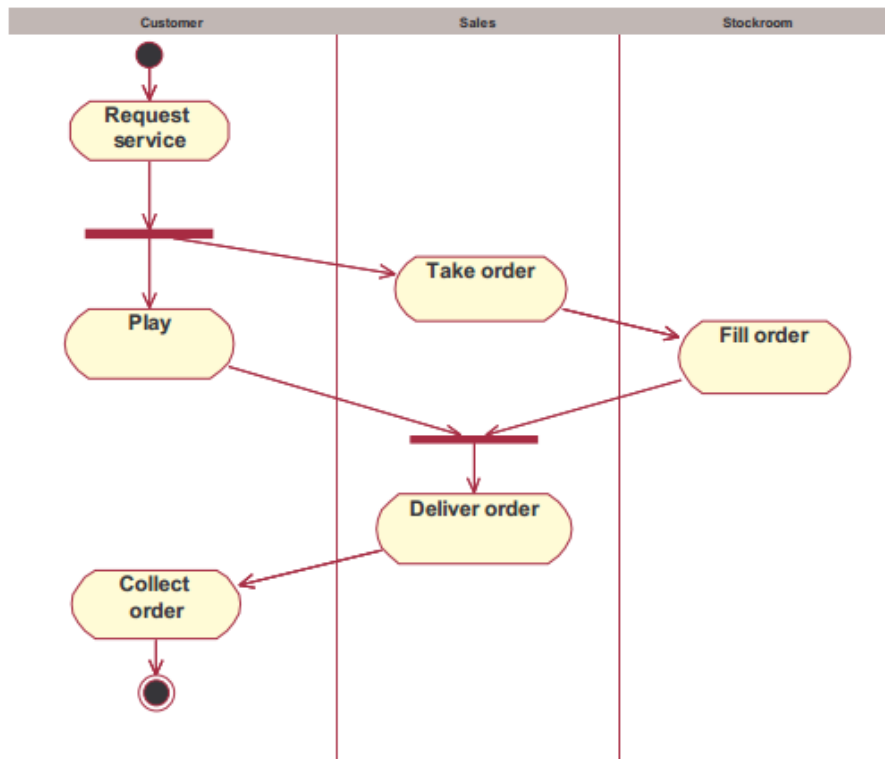
Un Diagrama de Colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre los mismos (en cuanto a la interacción se refiere).

Esta representación muestra a una serie de objetos con los enlaces entre los mismos, y con los mensajes que se intercambian dichos objetos. Los mensajes son flechas que van junto al enlace por el que “circulan”, y con el nombre del mensaje y los parámetros (si los tiene) entre paréntesis.

<sup>8</sup> [www.google.com/diagramadecolaboración/imágenes](http://www.google.com/diagramadecolaboración/imágenes)

#### 4.7. Diagrama de Actividades

Gráfica 4.8. Diagrama de actividades<sup>9</sup>



Fuente: [www.google.com/diagramadeactividades/imagenes](http://www.google.com/diagramadeactividades/imagenes). 2015

Puede visualizarse como un diagrama de estados personalizado en el cual casi todos los estados son estados acción y casi todas las transiciones evolucionan al término de dicha acción. Un diagrama de actividades puede dar detalle a un caso de uso, un objeto o un mensaje en un objeto.

<sup>9</sup> [www.google.com/diagramadeactividades/imagenes](http://www.google.com/diagramadeactividades/imagenes)



Un diagrama de actividad es utilizado en conjunción de un diagrama uso-caso para auxiliar a los miembros del equipo de desarrollo a entender como es utilizado el sistema y cómo reacciona en determinados eventos.

Los diagramas de actividades contienen los siguientes elementos:

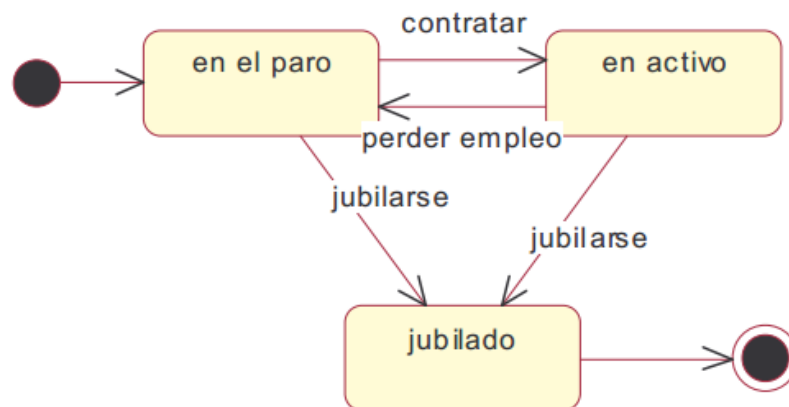
- Inicio: El inicio de un diagrama de actividad es representado por un círculo de color negro sólido.
- Actividad: Una actividad representa la acción que será realizada por el sistema la cual es representada dentro de un ovalo.
- Transición: Una transición ocurre cuando se lleva a cabo el cambio de una actividad a otra, la transición es representada simplemente por una línea con una flecha en su terminación para indicar dirección.

## 4.8 DIAGRAMA DE ESTADO

Un Diagrama de Estados muestra la secuencia de estados por los que pasa un caso de uso, bien un objeto a lo largo de su vida, o bien todo el sistema. En él se indican los eventos que son de un estado a otro y cuáles son las respuestas y acciones que genera.

En cuanto a la representación, un diagrama de estados es una representación gráfica en los que los nodos son estados y los arcos dirigidos son transiciones.

Gráfica 4.10. Diagrama de estado<sup>10</sup>



Fuente: <http://www.codecompiling.net/>. 2015

Un diagrama de estados puede representar ciclos continuos o bien una vida finita, en la que hay un estado inicial de creación y un estado final de destrucción (finalización del caso de uso o destrucción del objeto). El estado inicial se muestra como un círculo sólido y el estado final como un círculo sólido rodeado de otro círculo. En realidad, los estados inicial y final son pseudoestados, pues un objeto no puede “estar” en esos estados, pero nos sirven para saber cuáles son las transiciones iniciales y finales. (Perez, 2012)

---

Fuente: <http://www.codecompiling.net/>.

## **4.9 DIAGRAMA DE OBJETOS**

Los diagramas de objetos son utilizados durante el proceso de Análisis y Diseño de los sistemas informáticos. Se puede considerar un caso especial de un diagrama de clases en el que se muestran instancias específicas de clases (objetos) en un momento particular del sistema.

Los diagramas de objetos utilizan un subconjunto de los elementos de un diagrama de clase. Los diagramas de objetos no muestran la multiplicidad ni los roles, aunque su notación es similar a los diagramas de clase. (Bermón, 2014)

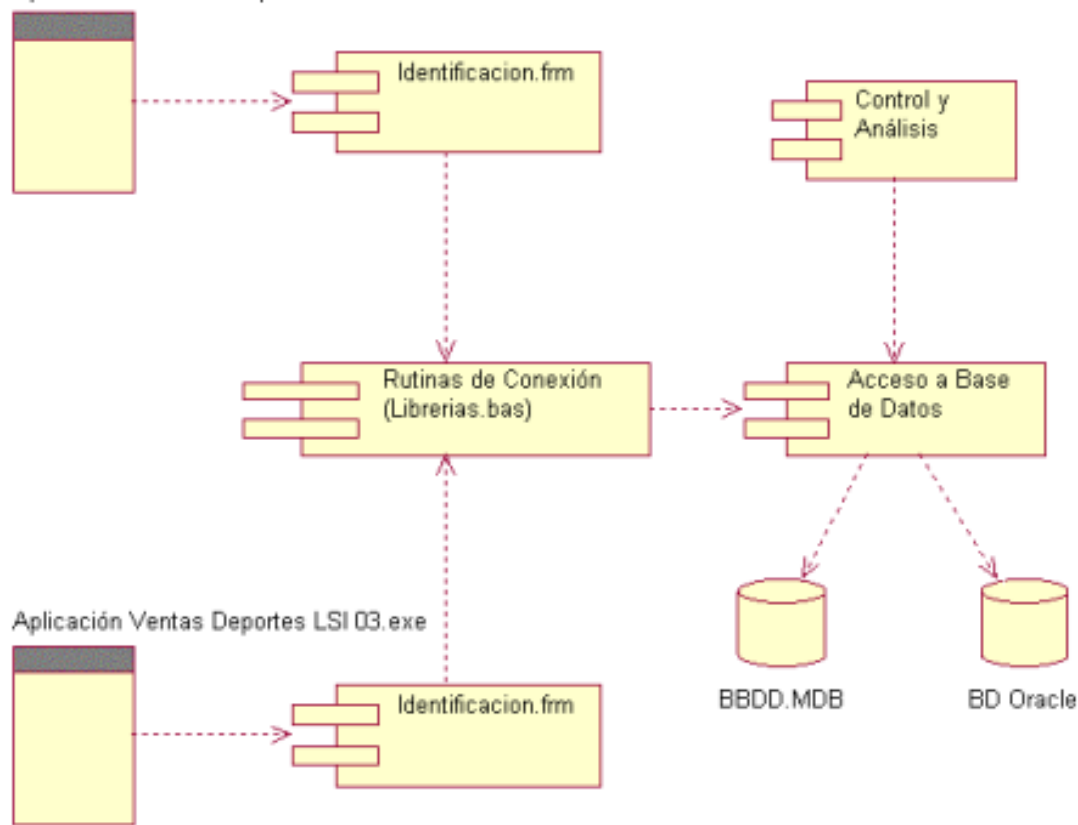
## **4.10 DIAGRAMA DE COMPONENTES**

Lo que distingue a un diagrama de componentes de otros tipos de diagramas es su contenido. Normalmente contienen componentes, interfaces y relaciones entre ellos.

Y como todos los diagramas, también puede contener paquetes utilizados para agrupar elementos del modelo.

Los componentes pertenecen a un mundo físico, es decir, representan a un bloque de construcción al modelar aspectos físicos de un sistema. Cada componente debe tener un nombre que lo distinga de los demás.

Al igual que las clases los componentes pueden enriquecerse con compartimientos adicionales que muestran sus detalles. (Perez, 2012)

Gráfica 4.11. Aplicación de modelo UML<sup>11</sup>

Fuente: <http://www.codecompiling.net/>. 2015

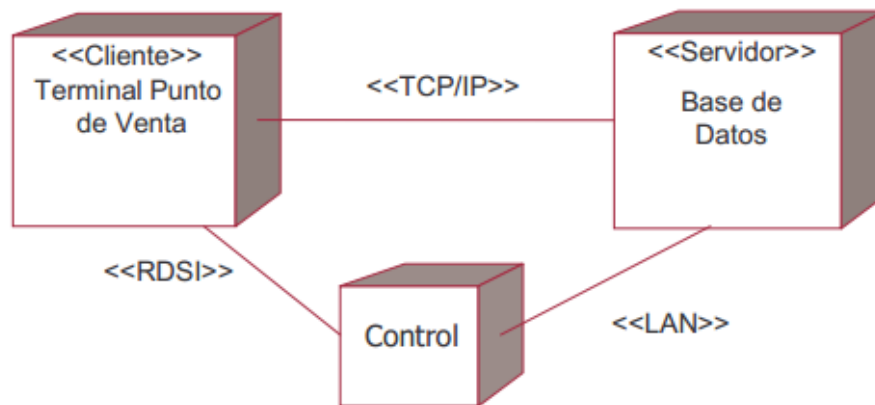
Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, etc. UML define cinco estereotipos estándar que se aplican a los componentes:

- **Executable:** Especifica un componente que se puede ejecutar en un nodo.
- **Library:** Especifica una biblioteca de objetos estática o dinámica.
- **Table:** Especifica un componente que representa una tabla de una base de datos.
- **File:** Especifica un componente que representa un documento que contiene código fuente o datos.
- **Document:** Especifica un componente que representa un documento.

<sup>11</sup> <http://www.codecompiling.net/>

#### 4.11 DIAGRAMAS DE DESPLIEGUE/DISTRIBUCION

Gráfica 4.12. Diagrama de despliegue/distribución<sup>12</sup>



Fuente: [www.google.com/diagramadedespliegueydistribución](http://www.google.com/diagramadedespliegueydistribución). 2015

Los Diagramas de Distribución muestran la disposición física de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos.

Un nodo es un elemento físico que existe en tiempo de ejecución y representa un recurso informático, que generalmente tiene algo de memoria y, a menudo, capacidad de procesamiento.

Los nodos se utilizan para modelar la topología del hardware sobre el que se ejecuta el sistema. Representa típicamente un procesador o un dispositivo sobre el que se pueden desplegar los componentes. (Wikipedia, 2013)

<sup>12</sup> [www.google.com/diagramadedespliegueydistribución](http://www.google.com/diagramadedespliegueydistribución)

En resumen:

El modelado permite la abstracción del sistema de software que buscará satisfacer y cumplir los requerimientos y necesidades del cliente.

UML implementa un modelado visual que se usa para especificar, visualizar, construir y documentar elementos de un sistema de software en un lenguaje común para los desarrolladores antes de generar líneas de código.

En el modelado UML se consideran tres aspectos importantes que son la descripción lógica de los elementos que interviene en el sistema, la interacción de dichos elementos y la infraestructura física en la que se desempeñan dichos elementos.

A continuación se muestra una tabla resumen de los distintos diagramas agrupados con el ámbito por el cual se aplica:

Tabla N.3 Matriz de conclusiones.

Área	Vista	Diagrama	Conceptos principales
Estructura	Vista estática	Diagrama de clases	Clase, asociación, generalización, dependencia, realización, interfaz
	Vista de casos de uso	Diagrama de casos de uso	Caso de uso, actor, asociación, extensión, inclusión, generalización del caso de uso
	Vista de implementación	Diagrama de componentes	Componente, interfaz, dependencia, realización
	Vista de despliegue	Diagrama de despliegue	Nodo componente dependencia, localización
Dinámica	Vista de máquinas de estados	Diagrama de estados	Estado, evento, transición, acción
	Vista de actividad	Diagrama de actividad	Estado, actividad, transición de terminación, división, unión

	Vista de interacción	Diagrama de secuencia	Interacción, objeto, mensaje, activación.
		Diagrama de colaboración	Colaboración, interacción, rol de colaboración, mensaje
Gestión del modelo	Vista de gestión del modelo	Diagrama de clases	Paquete, subsistema, modelo

## Capítulo V

### 5. Análisis de Requerimientos del Sistema

#### 5.1 Introducción

En base al problema inicial se procede a describir las funcionalidades que deberá disponer el sistema de software para satisfacer las necesidades de transporte de mercadería. Para el análisis de proyecto se basará en los siguientes criterios de requerimientos:

- Identificar las funcionalidades del sistema y detallarlos hasta hacerlos no ambiguos.
- Establecer el alcance de la aplicación, identificando los agentes externos con los que interacciona.
- Identificar las características de las interacciones.

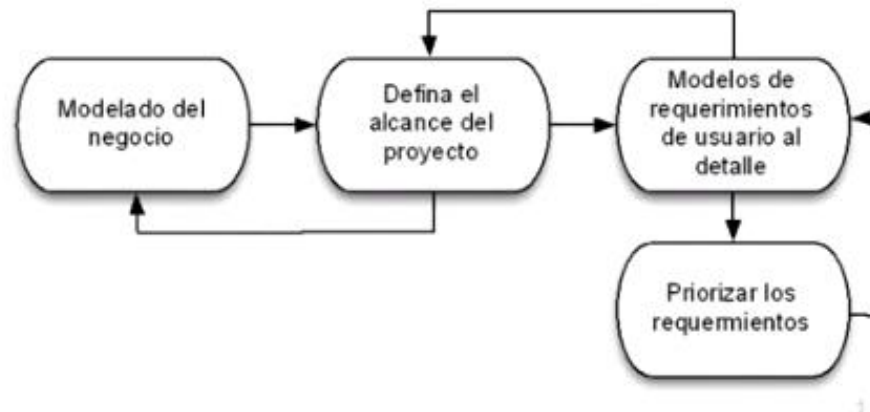
Bajo estos lineamientos se desarrollará el proceso de análisis de requerimientos como se muestra en la gráfica a continuación. Estos pasos serán fundamentales para el siguiente paso de diseño y modelado del sistema propuesto.

El modelado de negocio permitirá describir el campo de acción, los recursos e interacciones que se emplean en la organización y sobre la cual se aplicará el sistema.

La definición del alcance a través de los requerimientos funcionales y no funcionales permitirá obtener una meta clara del proyecto y un cronograma de trabajo claro.

Los modelos abstraen la realidad en un lenguaje común de trabajo y próximo al desarrollo de la codificación de la solución en un orden establecido por prioridades.



Gráfica 5.1. Diagrama de proceso análisis de requerimientos<sup>13</sup>

Fuente: [www.codecompiling.net/](http://www.codecompiling.net/). Elaboración Johan Cedeño. 2015

## 5.2 Sistema Actual

La empresa de transportes JARCARGO requiere de la interacción en tres diferentes niveles de usuarios o actores con la herramienta informática, para efectuar los procesos correspondientes al giro del negocio. Cada uno de estos niveles tiene un ámbito de trabajo independiente pero influyente sobre los otros procesos.

Los actores descritos tienen perfiles diferentes de uso del sistema y alimentan el sistema en distintos puntos de acuerdo al mapa de procesos de la Empresa.

Tipo de usuario	Administrador del Sistema
Descripción	Personal encargado de administrar los accesos y permisos a la herramienta de acuerdo al perfil.
Habilidades	Conoce el proceso logístico de la empresa y la herramienta informática.
Actividades	Controla los procesos y el uso eficiente de la herramienta. Administra el sistema y permite el acceso requerido de acuerdo al papel del usuario.

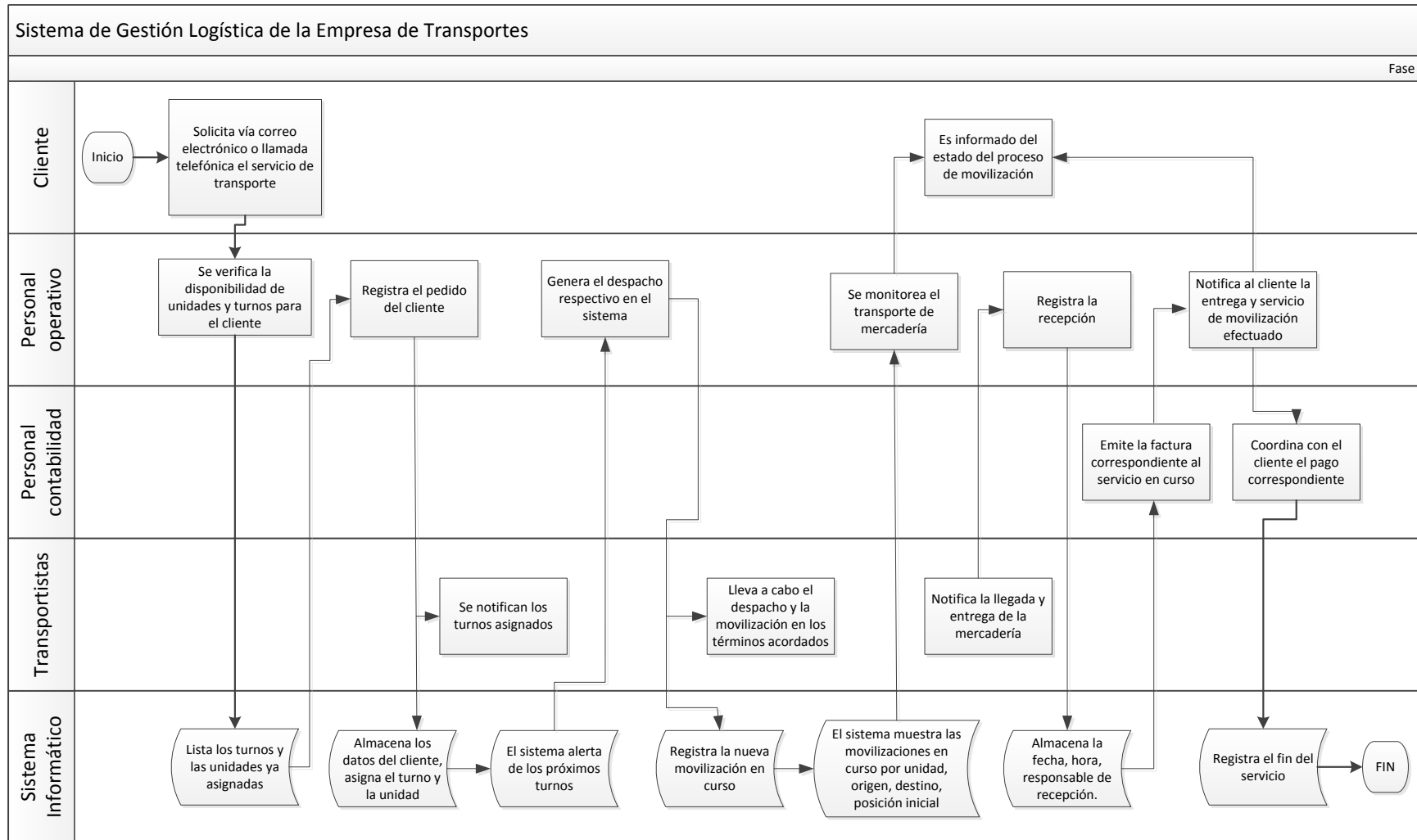
<sup>13</sup> [www.codecompiling.net/](http://www.codecompiling.net/)

Tipo de usuario	Personal de Operaciones
Descripción	Personal operativo que hace uso de la herramienta para gestionar los procesos de transporte de la empresa.
Habilidades	Conocimiento los procesos del negocio y del flujo de información que alimentan el sistema.
Actividades	Gestionar los procesos de registro de pedidos y contratos, despacho de mercadería, seguimiento de rutas y unidades, entrega de mercadería o mantenimiento de las unidades.

Tipo de usuario	Personal de contabilidad y cobranzas
Descripción	Personal administrativo contable que registra los procesos contables de las transacciones comerciales de los servicios prestados
Habilidades	Conocimiento de los procesos y documentos contables de la empresa.
Actividades	Gestionar los procesos de facturación o gestión de cobro. Reportes financieros, asignación de recursos.

Estos actores son el ente humano que gestiona los procesos de la empresa y que serán los usuarios del sistema informático. Cada proceso es secuencial y se alimenta de entradas para entregar un valor agregado al final. En el mapa o modelado de procesos, *Gráfica 5.2*, se muestra el flujo de actividades comunes en el servicio de transporte de mercaderías.

Gráfica 5.2. Modelado de negocio



Elaboración: Johan Cedeño. 2015

### 5.3 Sistema actual

JARCARGO actualmente no cuenta con una herramienta integral que automatice los procesos y flujos de información de contratos y gestión de movimiento de unidades.

El departamento logístico hace uso de hojas de cálculo en el que se registran los datos tomados inicialmente en formularios de papel. Estos documentos son registrados en los procesos de registros de pedidos, despachos y recepción de mercadería.

Sin embargo, el no tener una herramienta formal, ni disponer de información unificada con trazabilidad en tiempo real, abre la posibilidad de reprocesos, pérdida de información, ineficiencia del manejo de recursos y poca capacidad de visión del movimiento de los activos y unidades de trabajo empleadas en servicio.

#### Desventajas

- Información dispersa y con alta probabilidad de errores.
- Poca estandarización de la información transmitida entre departamentos.
- Altos tiempos de gestión de información para la toma de decisiones.
- Grandes esfuerzos para cruzar información y dar seguimiento de un servicio activo.

## 5.4 Sistema propuesto

Inicialmente se ha dispuesto levantar los requerimientos funcionales y no funcionales que requiere nuestro sistema para automatizar los procesos del mapa del negocio en una sola herramienta.

### 5.4.1 Requisitos funcionales del sistema de software

Un requisito es una condición necesaria para satisfacer una necesidad existente. Así los requisitos funcionales son declaraciones que debe proporcionar el sistema, de manera que den el valor agregado esperado a partir de entradas particulares, además este debe ser su comportamiento en situaciones particulares.

A partir del mapa de procesos se han identificado los siguientes requerimientos funcionales:

1. Gestión de credenciales de usuarios
2. Administración de cuentas de clientes
3. Gestión de solicitudes de servicio de transporte
4. Despacho de mercadería
5. Seguimiento del servicio de transporte
6. Entrega y destino final de mercadería
7. Facturación de servicio prestado
8. Gestión de cobro
9. Gestión de mantenimiento de las unidades

#### 5.4.1.1 Requisito funcional 1

Número de requisito	R1
Nombre de requisito	Administración de cuentas de clientes
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

El sistema debe permitir la generación de credenciales de los usuarios que operan los procesos y autenticar si se permite el acceso a la aplicación.

Gráfica 5.4.1 Ingreso de credenciales de usuarios del sistema<sup>14</sup>



Fuente: [www.google.com/imagenesdecredencialdeusuario/](http://www.google.com/imagenesdecredencialdeusuario/)

Tabla 4 Requisito funcional 1

ENTRADAS	PROCESOS	SALIDAS
Nombre de usuario, clave privada, datos de clientes.	El sistema validará el ingreso de las credenciales de un usuario. No validará la cuenta si se registra el campo vacío. La clave debe contener un mínimo de seguridad.	Autenticación válida. Mensaje de error en el caso de no haber llenado algún campo. Mensaje de error si no se encuentra al usuario. Mensaje de error si el usuario es correcto pero su clave es incorrecta. Mensaje de error si la cuenta de usuario no está activa.

Elaboración: Johan Cedeño. 2015

<sup>14</sup> Fuente: [www.google.com/imagenesdecredencialdeusuario/](http://www.google.com/imagenesdecredencialdeusuario/)

## 5.4.1.2 Requisito funcional 2

Número de requisito	R2
Nombre de requisito	Administración de cuentas de clientes
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

El sistema informático debe permitir la administración de las cuentas de los clientes a quienes se presta el servicio de transporte. El ingreso, mantenimiento y actualización de los clientes. Búsqueda y visualización de servicios activos por cliente.

Gráfica 5.4.2. Ejemplo de Formulario ficha de cliente<sup>15</sup>

The screenshot displays a web-based form for client information. The form is organized into several sections:

- Cliente (Casillas con ? son opcionales):** This section contains fields for personal identification and details. It includes a dropdown for 'Tipo de ID' (set to 'Cédula'), a text field for 'Número ID' (001-1234567-1), a dropdown for 'Masculino', and a date field for 'Estado Civil' (12/19/2089). Other fields include 'Apodo:?' (Nando), 'Ciudad de Nac.' (Dajabon), 'País de Nac.' (República Dominicana), 'Profesión' (Software Developer), and 'Nom. de Madre?' (Maria Batista). A small portrait photo of the client is shown on the right.
- Información Laboral y de Ingresos:** This section includes a table with columns for 'Empresa / Entrada', 'Ingreso', 'Estado', and 'Cobros'. The data row shows 'Softelk Corporation', '\$50,000.00', 'Empleado', and 'Mensual'.
- Direcciones:** This section includes a table with columns for 'Tipo' and 'Dirección'. It lists 'Hogar Propio' at 'Calle 6 número 34, Santiago De Los Caballeros, Santiago' and 'Laboral' at 'Calle 8 número 340, Santiago De Los Caballeros, Santiago'.
- Contactos Telefónicos:** This section includes a table with columns for 'Nombre', 'Tipo', 'Número', and 'Ext'. It lists 'Fernando Antonio Ramos Batista' with a home phone number '809-548-5487' and 'Softelk Corporation' with a labor phone number '809-567-5415'.
- Contactos Digitales:** This section includes a table with columns for 'Email/Cuentas'. It lists 'elEmail@emails.com'.
- Referencias:** This section includes a table with columns for 'Referente' and 'Tipo'. It lists 'Referente' with a labor type and 'Referente2' with a commercial type.

Fuente: [www.google.com/imagenes/formularioficha/](http://www.google.com/imagenes/formularioficha/)

<sup>15</sup> [www.google.com/imagenes/formularioficha/](http://www.google.com/imagenes/formularioficha/)

Tabla 5 Requisito funcional 2

ENTRADAS	PROCESOS	SALIDAS
Nombre de cliente, datos asociados a la cuenta como RUC, dirección, correo electrónico, contacto.	El sistema mostrará en una pantalla con el listado de clientes usuarios del sistema de transporte. Permitirá la creación de nuevas cuentas, la eliminación de estas y el seguimiento de los servicios activos pertenecientes a cada cuenta.	Listado de clientes registrados, servicios en curso asignados a los clientes registrados.

Elaboración: Johan Cedeño. 2015

#### 5.4.1.3 Requisito funcional 3

Número de requisito	R3
Nombre de requisito	Gestión de solicitudes de servicio de transporte
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

El sistema debe permitir el registro de solicitudes de servicio, asignación de turnos por cliente, asignación de rutas de acuerdo a las unidades disponibles de acuerdo a las necesidades en distancia y cantidad y tipo de mercadería a transportar; estimación de tiempos.

Gráfica 5.4.3 Diagrama de gestión de solicitudes de transporte<sup>16</sup>



Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

<sup>16</sup> [www.google.com/imágenes](http://www.google.com/imágenes)



Tabla 6 Requisito funcional 3

ENTRADAS	PROCESOS	SALIDAS
Fecha de inicio de transporte, fecha de entrega o fin de transporte. Lugar de partida, lugar de llegada, tipo y cantidad de mercadería a transportar, documentos que acompañan a la mercadería.	El sistema mostrará a través de un formulario el ingreso del pedido del cliente con sus datos	Turno asignado, unidad y personal asignada al despacho.

Elaboración: Johan Cedeño. 2015

#### 5.4.1.4 Requisito funcional 4

Número de requisito	R4
Nombre de requisito	Despacho de mercadería
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

El sistema debe permitir el registro del despacho y/o carga de mercadería en la unidad asignada e inicio del transporte desde el lugar de partida en la fecha de programada.

Gráfica 5.4.4 Despacho de mercadería<sup>17</sup>



Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

<sup>17</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

Tabla 7 Requisito funcional 4

ENTRADAS	PROCESOS	SALIDAS
Características de la mercadería que se transporta en volumen, cantidad y valor monetario. Número de factura o documento que respalda la mercadería a transportar. Lugar y fecha de inicio de transporte. Persona que conduce la unidad de transporte.	El sistema mostrará a través de un formulario el ingreso de los datos del despacho.	Guía de remisión que respalda el transporte de la mercadería.

Elaboración: Johan Cedeño. 2015

#### 5.4.1.5 Requisito funcional 5

Número de requisito	R5
Nombre de requisito	Seguimiento del servicio de transporte
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

En la aplicación se podrán visualizar las unidades de servicio en curso con los datos en tiempo real, a través de información tabulada y un mapa geográfico.

Gráfica 5.4.5. Diagrama de seguimiento de línea del servicio de transporte<sup>18</sup>



Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

<sup>18</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

Tabla 8 Requisito funcional 5

ENTRADAS	PROCESOS	SALIDAS
Numero de contrato, número de identificación del cliente	Se mostrará dinámicamente el despliegue de unidades en las vías del Ecuador. Se podrá buscar por cuadros de búsqueda datos específicos de un transporte en particular.	Mapa digital y datos de transportistas, mercaderías y tiempos pronosticados de arribo.

Elaboración: Johan Cedeño. 2015

## 5.4.1.6 Requisito funcional 6

Número de requisito	R6
Nombre de requisito	Entrega y destino final de mercadería
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

La aplicación permitirá el ingreso de los datos de arribo de una unidad de transporte, su estado y novedades en el desplazamiento.

Gráfica 5.4.6. Diagrama de entrega y conteo de mercadería<sup>19</sup>Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)<sup>19</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

Tabla 9 Requisito funcional 6

ENTRADAS	PROCESOS	SALIDAS
Numero de contrato, número de identificación del cliente, fecha y lugar de arribo, novedades del viaje, datos de conteo de la mercadería ingresada, guía de remisión que acompaña la mercadería.	El sistema mostrará a través de un formulario el ingreso de los datos que arriban al lugar destino.	Ninguna

Elaboración: Johan Cedeño. 2015

#### 5.4.1.7 Requisito funcional 7

Número de requisito	R7
Nombre de requisito	Facturación de servicio prestado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

La aplicación generará automáticamente la factura de los servicios prestados por el tiempo, distancia, cantidad transportada y unidad empleada para el transporte.

Gráfica 5.4.7. Diagrama de facturación de servicios prestados<sup>20</sup>



Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

<sup>20</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

Tabla 10 Requisito funcional 7

ENTRADAS	PROCESOS	SALIDAS
Ninguna. Los datos en base a los cuales se factura el servicio fueron ingresados	Imprimirá masivamente las facturas posteriores al arribo de la mercadería a su destino y la aceptación y conformidad del cliente. La generación de este documento se notificará al departamento de cobranzas.	Facturas emitida en el período actual. Notificaciones informativas para su correspondiente cobro.

Elaboración: Johan Cedeño. 2015

#### 5.4.1.8 Requisito funcional 8

Número de requisito	R8
Nombre de requisito	Gestión de cobro
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

La aplicación dará seguimiento de la facturas a cobrar, carteras vencidas y gestión de desempeño de la recuperación de valores.

Gráfica 5.4.8. Diagrama de gestión de cobranzas<sup>21</sup>



Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

<sup>21</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

Tabla 11 Requisito funcional 8

ENTRADAS	PROCESOS	SALIDAS
Números de contratos, número de identificación de clientes, fechas de búsqueda.	El sistema desplegará información tabulada de los servicios prestados en un periodo de tiempo, el estado respecto a su cobro, el historial de las transacciones registradas y controles de filtros para facilitar la búsqueda de información.	Clientes pendientes de cobro, valores adeudados, fechas e valores acumulados por intereses.

Elaboración: Johan Cedeño. 2015

#### 5.4.1.9 Requisito funcional 9

Número de requisito	R9
Nombre de requisito	Gestión de mantenimiento de las unidades
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

La aplicación permitirá la gestión de mantenimiento de unidades ya sea preventivo o correctivo para disponer una flota de unidades de listas para prestar el servicio.

Gráfica 5.4.9. Diagrama de gestión de mantenimiento de unidades<sup>22</sup>



Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

<sup>22</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

Tabla 12 Requisito funcional 9

ENTRADAS	PROCESOS	SALIDAS
Información de unidades con averías (placas de la unidad, kilometraje de recorrido, año de fabricación, daños registrados.)	Se desplegará una tabla de unidades con su respectivo estado y disponibilidad.	Notificaciones de unidades próximas a mantenimiento. Registro de unidades disponibles, no disponibles y dadas de baja.

Elaboración: Johan Cedeño. 2015

### 5.5 Requisitos no funcionales del sistema de software

Los requisitos no funcionales son aquellas carencias implícitas o propiedades emergentes que sin ellas el usuario no percibiría como satisfecha la necesidad de hacer uso de la herramienta en cuestión. A continuación, los requerimientos encontrados por el usuario.

### 5.6 Requisitos de rendimiento

La infraestructura de red, así como los terminales de los usuarios, deben cumplir con normas según la IEEE en la conexión a los equipos, para tener tiempos de respuesta mínimos.

Numero de terminales a manejar:

Se contará con un servidor de base de datos y de aplicaciones en la matriz de la empresa.

Número de usuarios simultáneos:

El número de usuarios máximo, que interactuaran simultáneamente con el sistema para la visualización de resultados, se diseñará con el analista.

Número de transacciones a manejar dentro de ciertos períodos de tiempo:

Se estima que se manejará alrededor de 10 transacciones de registro de pedidos diarios.

## **5.7 Seguridad**

La seguridad del sistema es por:

Uso de contraseñas para cada usuario. Esto permitirá que tengan acceso al sistema solo las personas que tienen autorización.

## **5.8 Fiabilidad**

Es uno de los factores que dará confianza al cliente, para lo cual el sistema está controlando todo tipo de transacción y está apto a responder todo tipo de incidente.

## **5.9 Disponibilidad**

El sistema será desarrollado para una disponibilidad de un 95%. De esta manera permitirá el desempeño de labores internas de la empresa.

## **5.10 Mantenibilidad**

El sistema contará con características parametrizables para los futuros mantenimientos. Se dispone realizar un mantenimiento preventivo en la etapa de pruebas y correctivo luego de la puesta en producción, en el caso de que existan errores.

El mejoramiento y nuevas implementaciones sobre el aplicativo incurrirán en costos para el cliente.

## **5.11 Portabilidad**

Hay ventajas al utilizar herramientas y lenguajes que garantizan la portabilidad. De esta manera:

- 99.9% de la aplicación es portable, por el simple hecho de utilizar un lenguaje y plataforma libre.
- 99% de la base de datos es portable, al utilizar un motor de base de datos relacional.

## **5.12 Pseudorequerimientos o Restricciones**



Se desarrollará una aplicación en la nube para a través de internet brindar una alta disponibilidad de servicio y una alta cobertura para su utilización en distintas sucursales de la empresa.

Se requiere una estructura de servidores que hospede el sitio de la aplicación web y la base de datos; además del dominio para que el sitio web pueda ser publicado en la Internet.

Adicionalmente, se precisa de un sistema de contingencia que respalde la información constantemente y pueda estar disponible para levantarse inmediatamente si el sitio principal se cae por cualquier eventualidad.

Grafica 5.12.1 Pseudorequerimientos<sup>23</sup>



Fuente:<http://3.bp.blogspot.com>

---

<sup>23</sup> <http://3.bp.blogspot.com>

## 5.13 Modelos del Sistema

### 5.13.1 Herramienta de Modelado

“Las técnicas de modelado gráfico son de particular interés, porque que son los más adecuados para los propósitos de comunicación y visualización. Las herramientas de modelado, permiten crear un "simulacro" del sistema, a bajo coste y riesgo mínimo. Además minimizan los riesgos, permiten concentrarse en ciertas características importantes del sistema, prestando menos atención a otras.”

Gráfica 5.13.1. Logotipo Enterprise Architect<sup>24</sup>



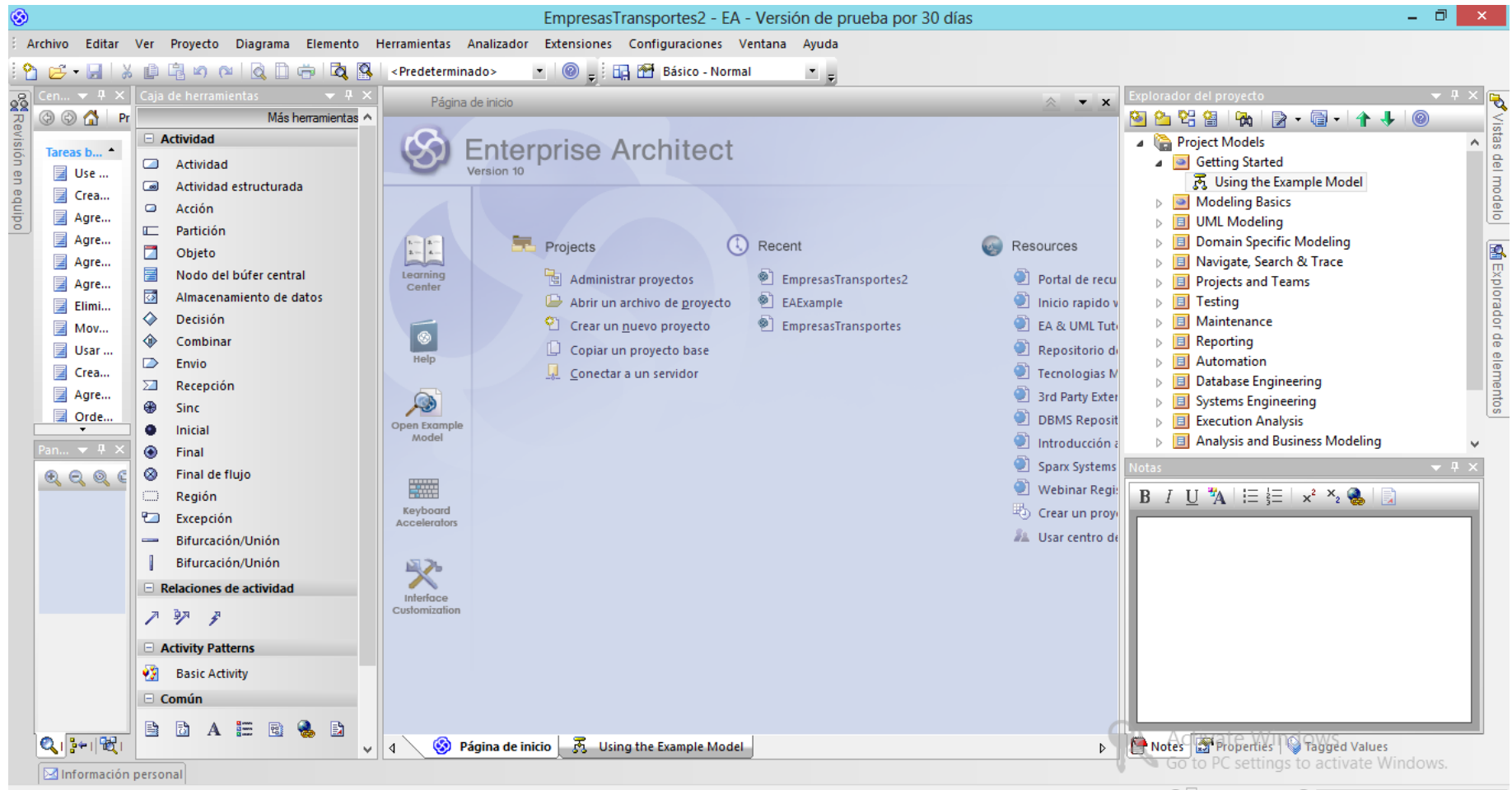
Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

Como instrumento para el modelado y diseño de la herramienta se utilizará Enterprise Architect. “Enterprise Architect es una plataforma de alto desempeño para el modelado, visualización y diseño, basada en el estándar UML 2.4.1. Ofrece trazabilidad completa desde mapas mentales, pasando por los requerimientos y hasta el diseño y la distribución del software, con el nivel de eficiencia, robustez, herramientas de colaboración y seguridad requeridos para sacar adelante proyectos altamente demandantes y cualquier tamaño. Ofrece salida de documentación flexible y de alta calidad. Algunas de sus características son: Diseño y construcción de UML, Casos de Uso, Modelos Lógico, Dinámico y Físic.” (www.unitty.com, 2014)

---

<sup>24</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

Grafica 5.13.2 Pantalla de inicio Enterprise Architect<sup>25</sup>



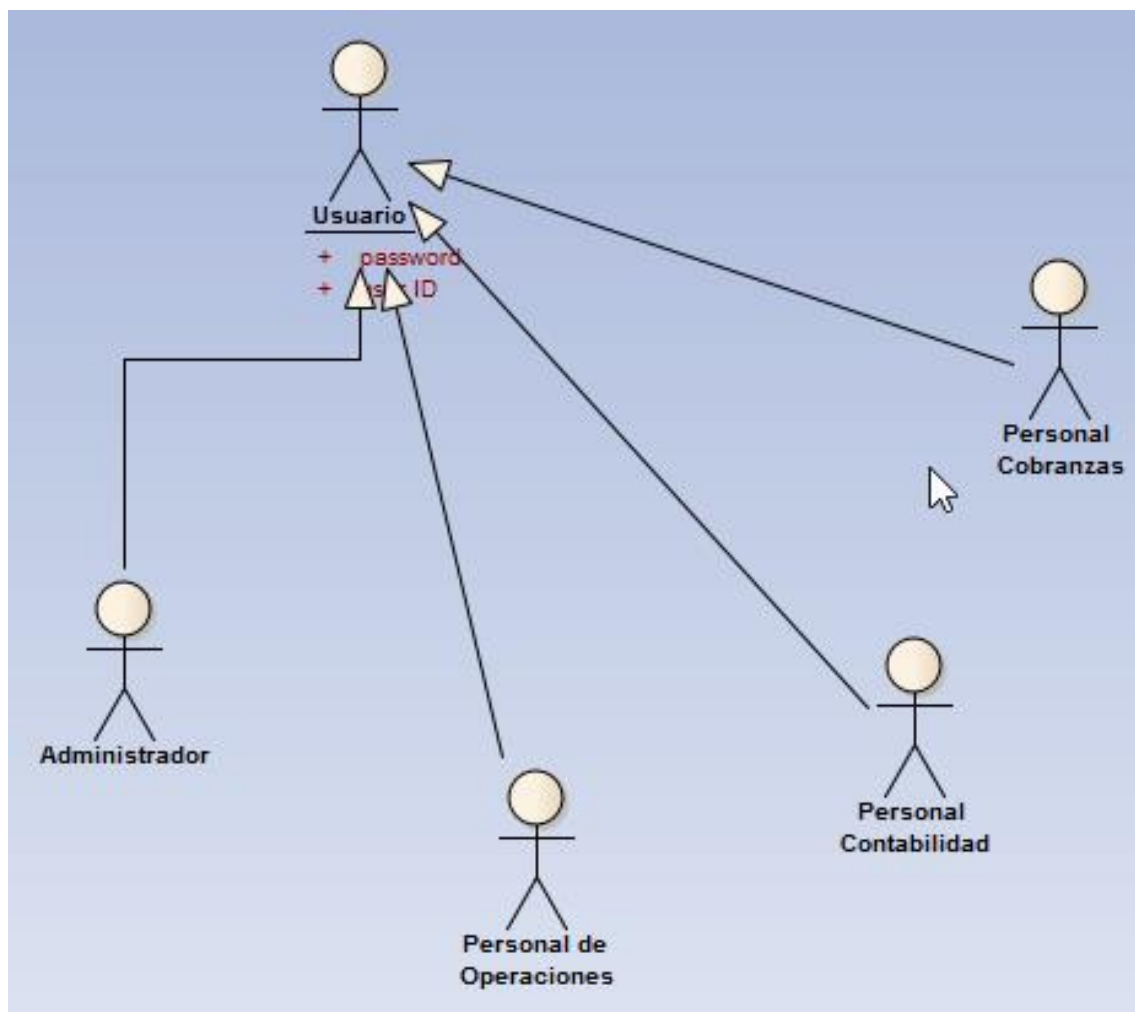
Fuente: Programa Enterprise Architect. 2015

<sup>25</sup> Programa Enterprise Architect

## Modelos de Casos de Uso

Diagrama de actores.- Muestra los tipos de usuarios representados por un actor genérico USUARIO que dispone de credenciales id y contraseña para ingresar al sitio.

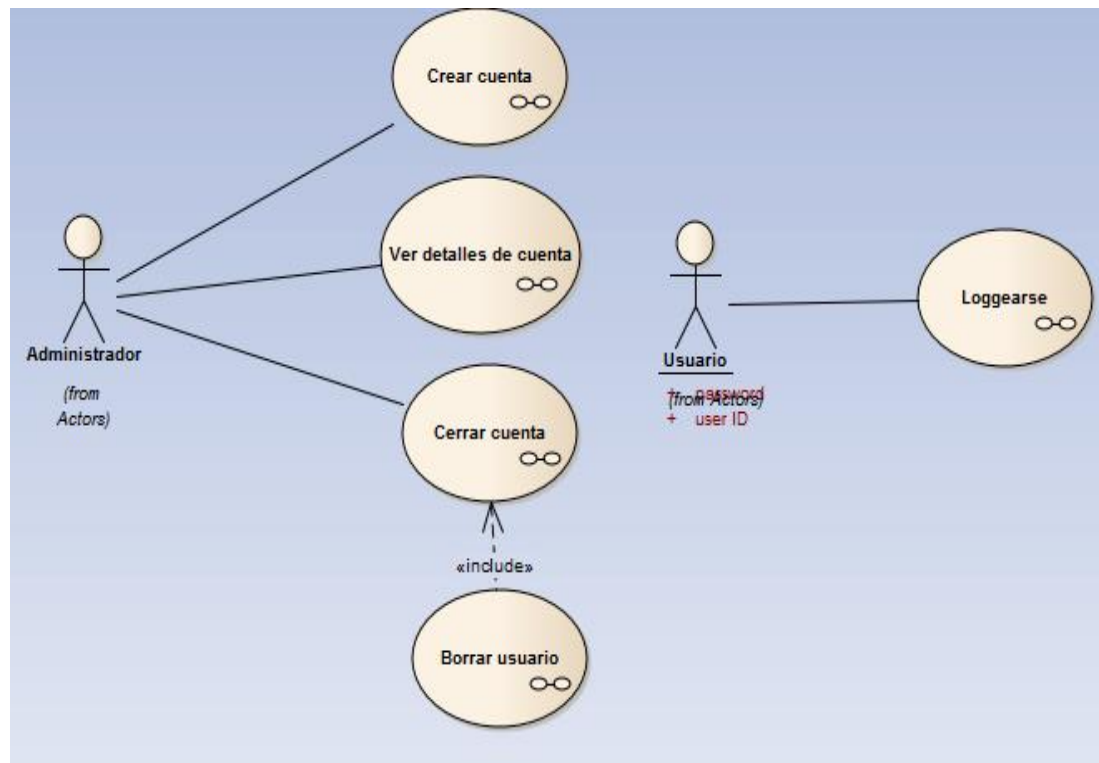
Gráfica 5.13.3 Diagrama de actores



Elaboración: Johan Cedeño. 2015

Gestión de credenciales de usuarios.- El actor Administrador es el perfil que tendrá la capacidad de crear, monitorear y eliminar las cuentas de usuarios del sistema. El actor genérico usuario podrá ingresar al sistema a través de sus credenciales.

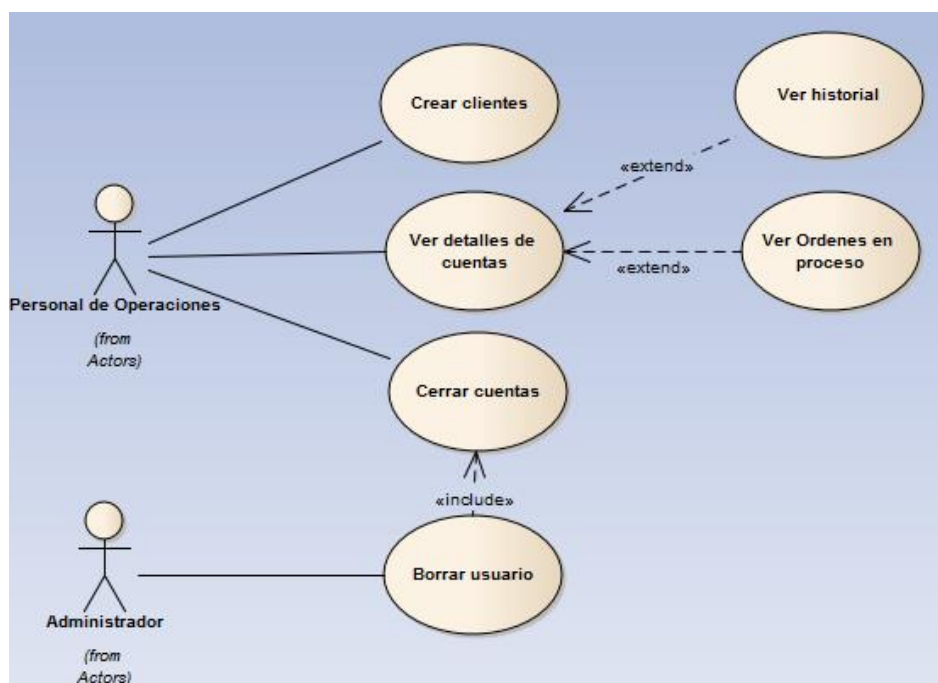
Gráfica 5.13.4. Gestión de credenciales de usuario



Elaboración: Johan Cedeño. 2015

Administración de cuentas de clientes.- Los usuarios con perfil de personal de operaciones pueden crear las cuentas de clientes en el sistema, monitorear sus datos y la información relacionada a sus pedidos, y desactivar las cuentas. Pero solo el administrador podrá borrar una cuenta.

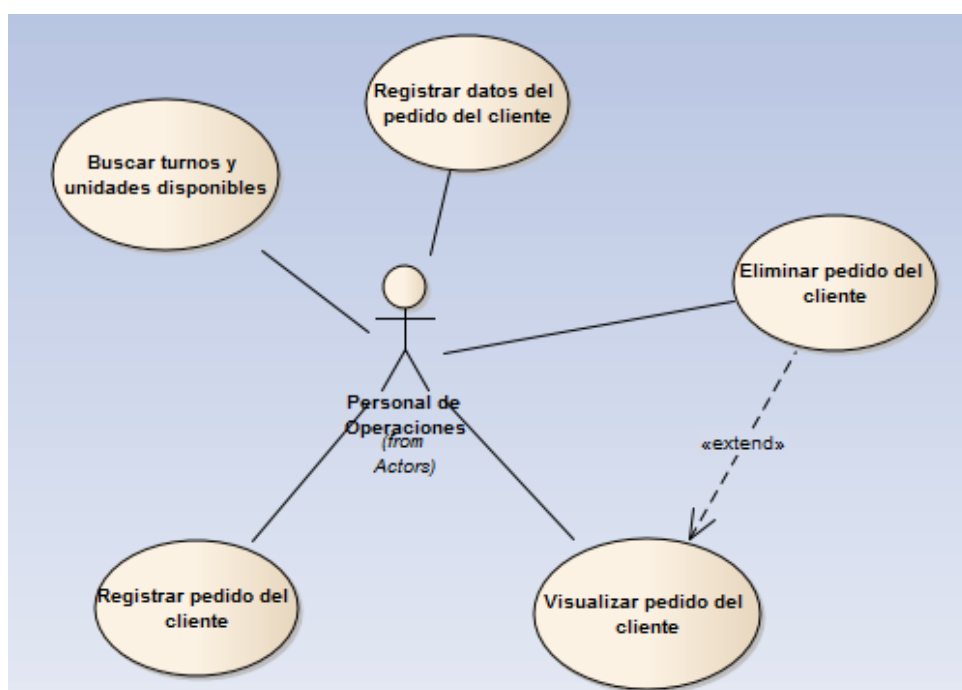
Gráfica 5.13.5. Administración de cuentas de clientes



Elaboración: Johan Cedeño. 2015

Gestión de solicitudes de servicio de transporte.- El perfil de personal de operaciones está habilitado para gestionar los procesos de pedidos desde crear uno, dar seguimiento, cerrarlo y eliminarlo.

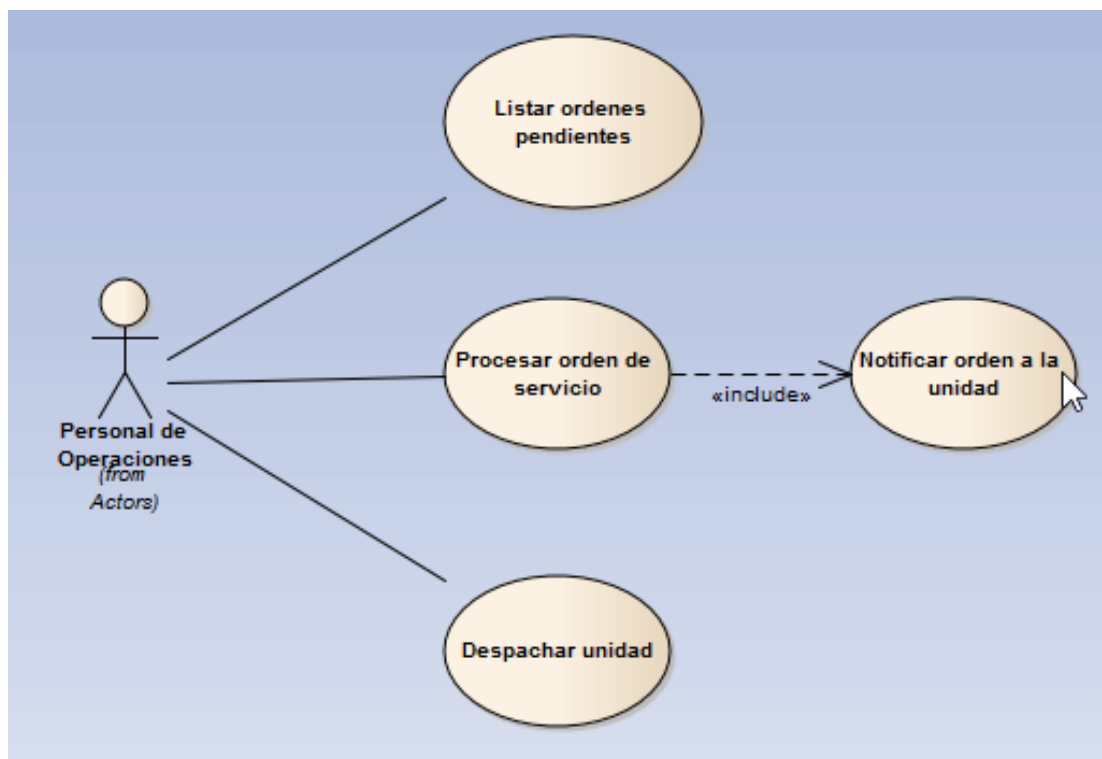
Gráfica 5.13.6. Gestión de solicitudes de servicio



Elaboración: Johan Cedeño. 2015

Despacho de mercadería.- El proceso de despacho y transporte de mercadería en un pedido está gestionado por el personal de operaciones.

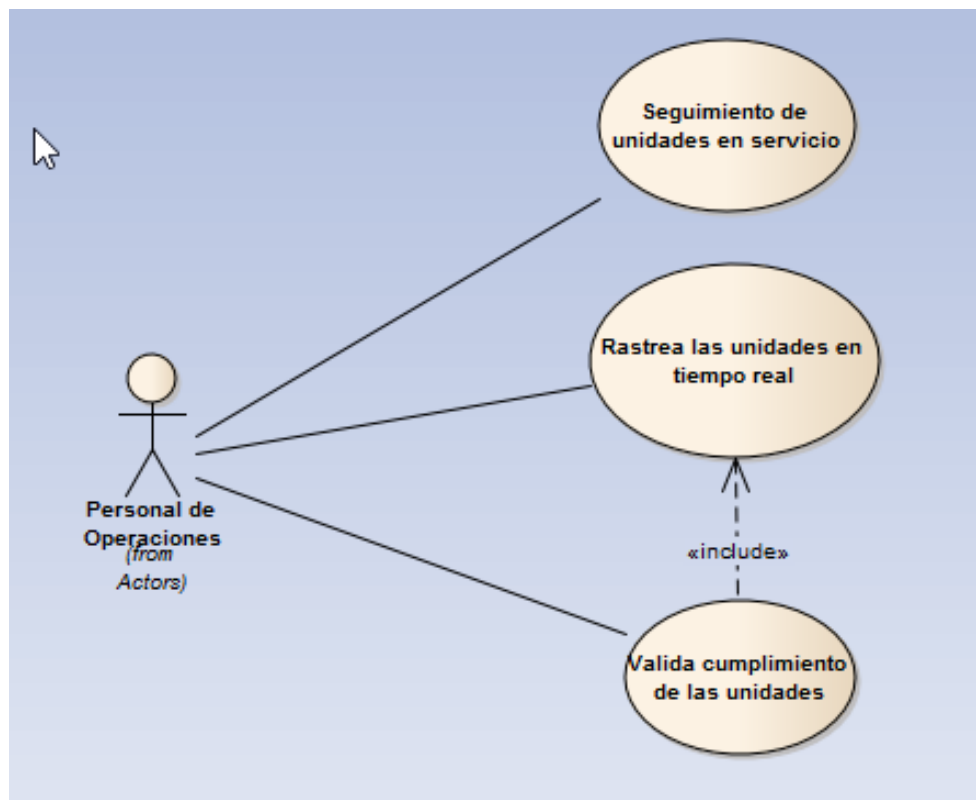
Gráfica 5.13.7. Despacho de mercaderia



Elaboración: Johan Cedeño. 2015

Seguimiento del servicio de transporte.- El seguimiento del pedido y las acciones de la unidad de transporte durante la movilización de la mercadería están a cargo del perfil de operaciones.

Gráfica 5.13.8. Seguimiento del servicio de transporte

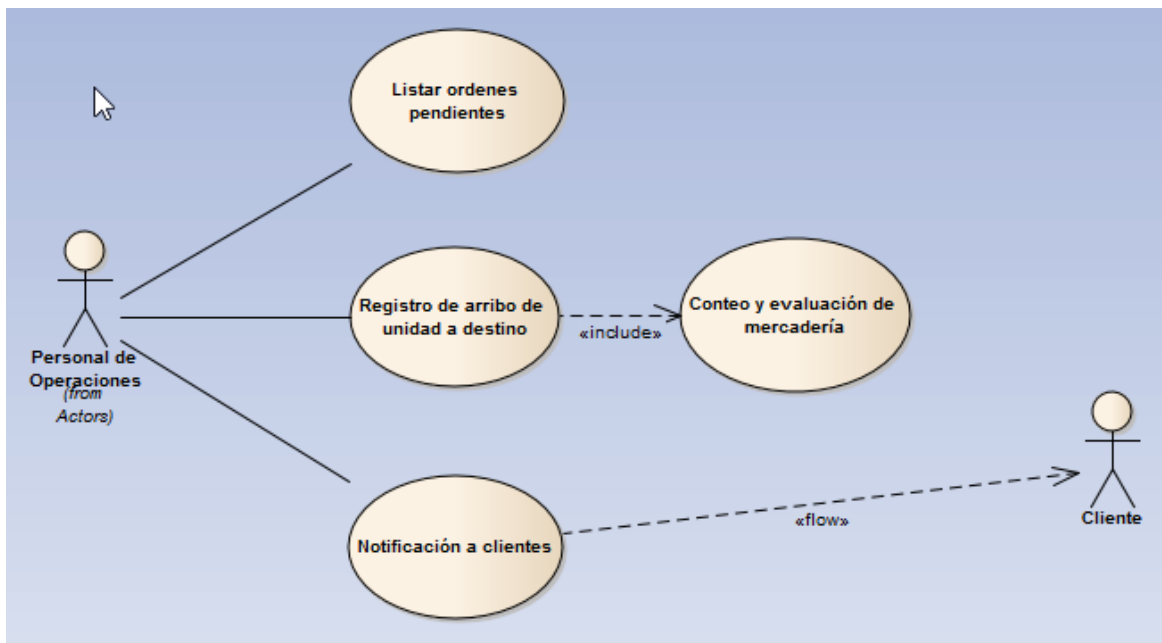


Elaboración: Johan Cedeño, 2015



Entrega de mercadería.- El proceso de entregar la mercadería en su destino final, el conteo de los ítems, y la verificación del estado de los mismos es recibida por el agente o transportista para ser ingresado en la aplicación con el perfil de operaciones. El cierre del proceso notificará automáticamente al cliente a través de un correo electrónico.

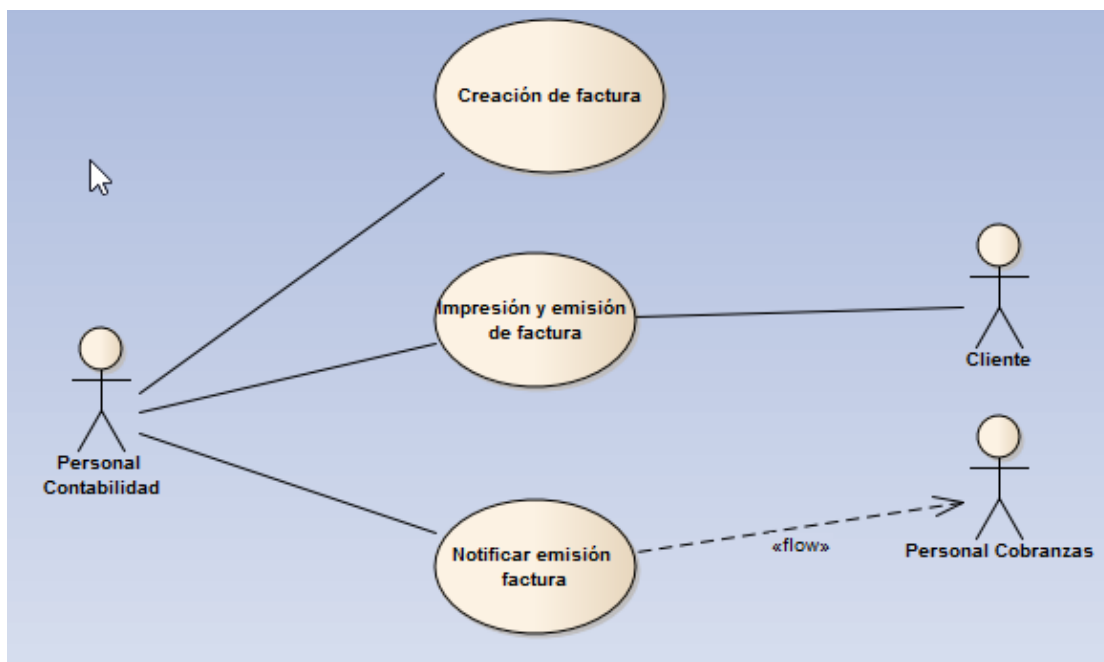
Gráfica 5.13.9. Entrega de mercadería



Elaboración: Johan Cedeño. 2015

Facturación de servicio prestado.- La generación de las facturas se producen luego de la entrega del pedido y es efectuada por el perfil de contadores. La emisión de los documentos es notificado al perfil de cobranzas para que estos a su vez continúen con el proceso de cobros.

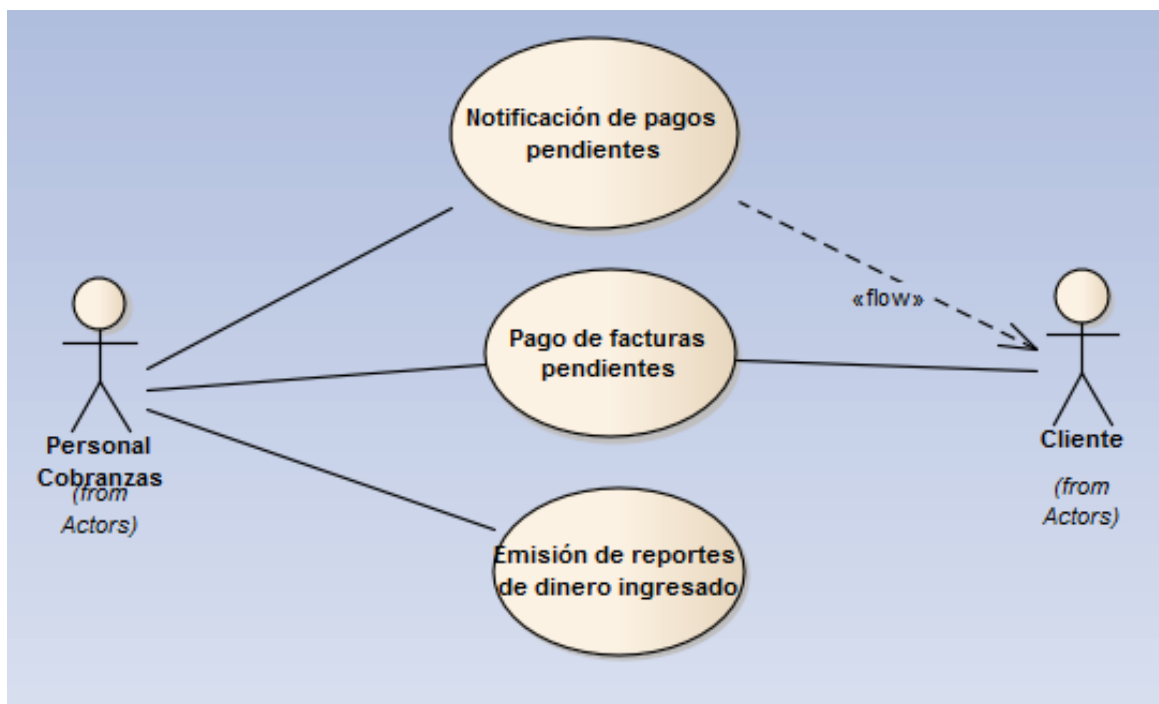
Gráfica 5.13.10. Facturación de servicios de cobro



Elaboración: Johan Cedeño. 2015

Gestión de cobro.- La recuperación de cuentas por cobrar la ejecuta el perfil de cobranzas y tiene comunicación directa con los clientes a través de notificaciones automáticas a los correos de los clientes.

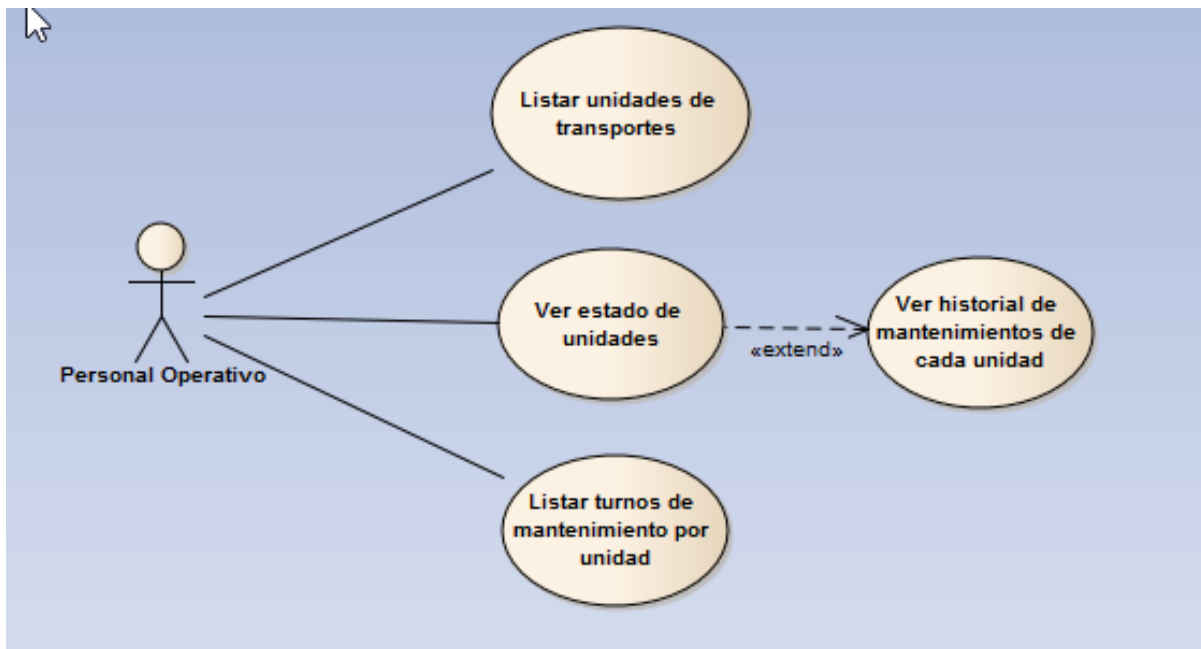
Gráfica 5.13.11 Gestión de cobro



Elaboración: Johan Cedeño. 2015

Gestión de mantenimiento de las unidades.- El proceso de mantenimiento de las unidades de transporte es efectuada por el personal de operaciones y dispone de acciones de gestión como monitorear las unidades, su cronograma de mantenimiento y los registros de los estados de cada una.

Gráfica 5.13.12. Gestión de mantenimiento de unidades



Elaboración: Johan Cedeño. 2015

En resumen:

El levantamiento de requerimientos describe las funcionalidades explícitas e inherentes que debe disponer el sistema de software para satisfacer las expectativas del cliente basados en el mapa de procesos de negocio de la compañía.

La empresa no dispone actualmente de una herramienta integral que automatice los procesos y flujos de información para un óptimo desempeño de las actividades y empleo de recursos en el servicio de transporte de mercaderías.

La información que se genera en las distintas etapas del mapa de procesos y que actualmente se encuentra dispersa se integrará y convergerá en una sola herramienta disponible en la nube, lo que mejorará la visualización del estado de servicio y la perspectiva de toma de decisiones.

Para la elección del paquete de diseño y modelado, se consideró varias características entre ellas: que implementen especificaciones UML 2.0, que se pueda integrar fácilmente con entornos de desarrollo como Eclipse y que disponga de un entorno de trabajo bastante intuitivo. Finalmente entre los paquetes más atractivos se eligió entre Enterprise Architect y Visual Paradigm for UML. Sin embargo el primero tiene mejores prestaciones, es una potente

multiplataforma que dispone de módulos para integración con BPMN, SoaML, SOMF, WSDL, XSD, TOGAF, entre otros; además de compiladores y simuladores. Y si bien es preciso disponer de una licencia comercial, su costo vale la pena para los desarrollos de los diagramas para diseño, documentación y codificación directa e inversa.

Tabla N 13. Análisis de herramientas del sistema

Elaboración: Johan Cedeño. 2015

Herramienta	Código abierto	Tipo de licencia	Lenguaje de prog utilizado	Integración con IDEs	Coste	UML2	Soporte para MDA	Soporte para XMI	Generación de código	Sistema Operativo
ArgoUML	Si	EPL	Java	NA	Gratuita	No	No	Si	C++, C#, Java, PHP4, PHP5, Ruby	Multiplataforma (Java)
Poseidon for UML	No	Comercial	Java	NA	Gratuita	NA	NA	NA	NA	Multiplataforma (Java)
Visual Paradigm for UML	No	Comercial, Community Edition	Java	Eclipse, NetBeans, IntelliJ and Visual Studio	Gratuita (version community)	Si	NA	Si (Version Comercial)	Java, C# (binary), C++, PHP	Multiplataforma (Java)
StarUML	No	Commercial	Delphi	C#	Gratuita (version académica)	Si	Si	Importar	Java Profile, C++ Profile, C# Profile Code Generator and Reverse Engineer	Windows, OS X
Rational Software Modeler	No	IBM EULA	Unknown	Eclipse	Gratuita	Si	Si	NA	NA	Windows, Linux
Enterprise Architect	No	Commercial	C++	Eclipse & Visual Studio	Gratuita (version trial) \$135 Version Desktop	Si	Si	Si	ActionScript, C, C#, C++, Delphi, Java, PHP, Python, Visual Basic, Visual Basic .NET, DDL, XML Schema, WSDL	Windows (Soporta instalación en Linux & Mac)
Umbrello UML Modeler	Si	GPL	C++, Java, Perl, PHP, Python	KDE	Gratuita	No	Si	Si	C++, IDL, Pascal/Delphi, Ada, Python, Java; import XML, RoseMDL	Unix, Windows

## Capítulo VI

### 6. Diseño del Sistema

#### 6.1 Introducción

La etapa de diseño se ocupará de estructurar los módulos del sistema apropiados para satisfacer los requerimientos realizados en el levantamiento de información y análisis de la solución.

El objetivo es desarrollar un sistema de información para el manejo de las operaciones de la empresa de transporte JARCARGO S.A. Este sistema responderá en línea a las peticiones de distintos puntos físicos o sucursales de la organización desde los que se acceda. Para ello se implementará una estructura Web, Gráfica 6.1, para su permanente acceso a través de la Internet.

Gráfica 6.1.1 Estructura de sistema web<sup>26</sup>



Fuente: <http://www.sparxsystems.com.ar/websistem>. 2015

La alta disponibilidad de servicio brindará flujos de datos demandados por los procesos de la empresa.

---

<sup>26</sup> <http://www.sparxsystems.com.ar/websistem>.

La información será almacenada en una base de datos centralizada que alimente las peticiones del sistema y módulos o sistemas paralelos como un ERP (planificación de recursos empresariales, por sus siglas en inglés, enterprise resource planning), sistema contable, CRM (administración basada en la relación con los clientes, sigla del término en inglés, customer relationship management) entre otros.

Esta estructura pretende satisfacer la necesidad de un sistema de información que contemple los requerimientos funcionales descritos en el capítulo anterior a través de las sucursales ubicadas en distintas ciudades del país y futuros puntos en proyección del negocio creciente.

Además, un diseño del sistema de información basado en n-capas, permitirá interactuar con clientes ligeros como navegadores de internet, teléfonos inteligentes, tabletas y muchos otros dispositivos preparados para conectarse a Internet.

## **6.2 Arquitectura del Sistema Actual**

El sistema informático que dispone la empresa es un aplicativo contable desarrollado a la medida. El sistema se halla distribuido en instancias de aplicaciones de escritorio instaladas en máquinas de Windows XP 32 bits. El servidor corre sobre un sistema operativo Windows Server 2000 y base de datos FoxPro. Actualmente no dispone de un sistema informático que integre el proceso operativo de gestión de pedidos, seguimientos de mercaderías y mantenimiento de unidades.

## **6.3 Arquitectura del Software Propuesto**

Se desarrollará una aplicación en la nube para a través de internet brindar una alta disponibilidad de servicio y una alta cobertura para su utilización en distintas sucursales de la empresa. Al ser una aplicación web permitirá centralizar el mantenimiento y actualizaciones del sistema y una diversificación de dispositivos desde los cuales se puede acceder.



### 6.3.1. Plataforma

Para la elección de la plataforma sobre la que correrá el sitio se toma en cuenta una de código abierto, debido a su costo, disponibilidad de información en la comunidad tecnológica y la portabilidad que ofrece.

Por su robustez en plataformas de sistemas distribuidos, core de servicios, módulos de seguridad y escalabilidad se sugiere desarrollarla sobre JAVA. Además de que permite ejecutar su código en cualquier sistema operativo por su máquina virtual. A más de ser una plataforma libre dispone de gran variedad de herramientas de código abierto e información de respaldo.

Gráfica 6.3.1.1 Logotipo JAVA<sup>27</sup>



Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

Para Java el servidor de aplicaciones web se plantea JBoss. Este servidor de aplicaciones dispone de versiones bastante estables con recursos de seguridad y estructura relativamente fácil de ampliar en infraestructura de hardware en el caso que se requiera de incrementar su demanda.

Gráfica 6.3.1.2 Logotipo JBoss<sup>28</sup>



---

<sup>27</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

<sup>28</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

Como motor de base de datos relacional se utilizará MySQL, siendo este motor bastante estable, dirigido por una comunidad de desarrolladores y ampliamente utilizado en una arquitectura sobre JBoss y JAVA como plataforma.

Gráfica 6.3.1.3. Logotipo MySQL<sup>29</sup>



Fuente: [www.google.com/imágenes](http://www.google.com/imágenes)

### 6.3.2. Arquitectura Lógica

En la arquitectura del sistema se describen los módulos de almacenes de datos, lógica de negocio, control de flujo y presentación; apilados de manera secuencial y conocidos cada uno como capa lógica del sistema.

- **Almacenamiento de datos.**- es la capa lógica que describe el registro de los datos de información ingresados a través de las interfaces de entrada o generados en las transacciones internas del sistema. Estos datos organizados como entidades estructuradas que se relacionan entre sí dan lugar a los procesos y representaciones del giro de negocio.
- **Lógica del negocio.**- es la capa que describe las reglas del negocio, la manera que se relacionan los datos, el flujo de datos y las validaciones que ordena la secuencialidad de los procesos del sistema.

---

<sup>29</sup> [www.google.com/imágenes](http://www.google.com/imágenes)

- **Capa de presentación y del cliente.**- Son los recursos informáticos que permiten el diálogo con el usuario del sistema, generalmente utilizan medios visuales de salida y periféricos de terminales para la entrada de datos. Constituye el medio de acción y producto final del sistema del proyecto.

En esta etapa inicial es importante diseñar la arquitectura del sistema en su totalidad implementando las funcionalidades requeridas y la estructura adecuada para que a futuro se implementen posibles mejoras, normalmente requeridas como crecimiento de capacidad, disposición de nuevos servicios e interconexión con otros sistemas o implementación de sistemas satélites.

Gráfica 6.3.2.1. Diagrama oracle<sup>30</sup>

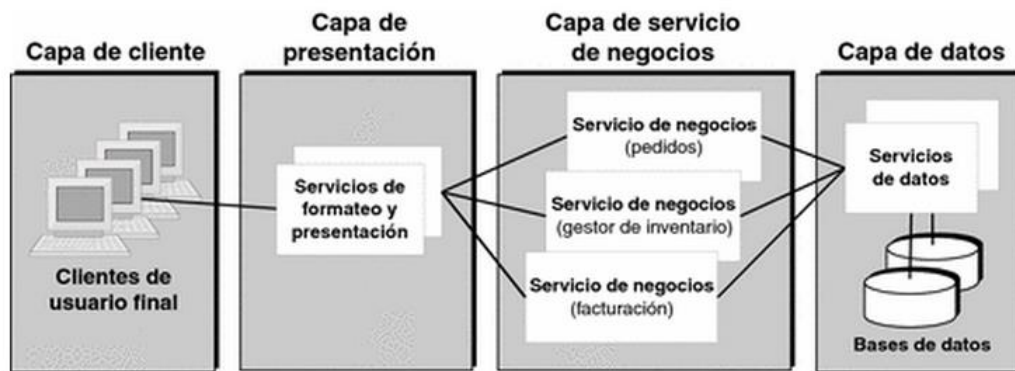


Diagrama según Oracle – Sun Java Enterprise Systema

<https://docs.oracle.com/cd/E19528-01/820-0888/auto22/index.html>

#### 6.3.2.1. Almacenamiento o Persistencia.

Es la capa lógica que se encarga de almacenar, gestionar y garantizar el acceso a los datos del sistema fundamentado en las propiedades de la información de atomicidad, consistencia, aislamiento y durabilidad. Para el almacenamiento se implementará una base de datos relacionada. El modelamiento y descripción se mostrará en el siguiente capítulo. En la capa de persistencia se utilizarán técnicas avanzadas de mapeo de datos relacionales a través módulos ORM que serán el motor de persistencia orientada a objetos.

<sup>30</sup> [docs.oracle.com/cd/E19528-01/820-0888/auto22/index.html](https://docs.oracle.com/cd/E19528-01/820-0888/auto22/index.html)

Para el diseño de la capa de almacenamiento es importante considerar los siguientes parámetros: <http://lapastillaroja.net/2013/07/app-escalable/>

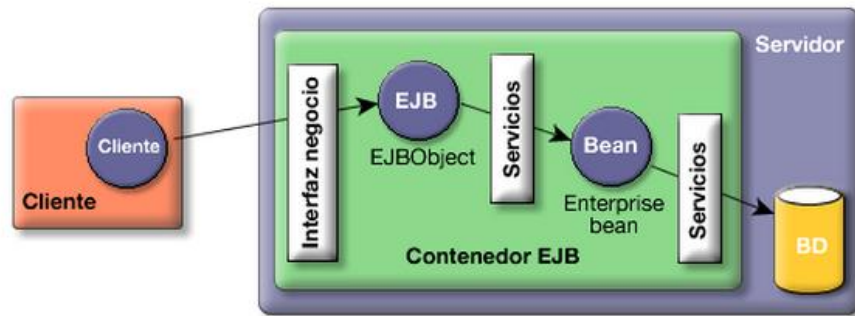
- volumen de datos
- número de escrituras en disco por unidad de tiempo
- número de lecturas en disco por unidad de tiempo
- complejidad de las consultas ejecutadas contra la base de datos.

Según la compatibilidad de tecnologías aplicadas a JAVA entorno a herramientas de código libre se propone como base de datos relacional a MySQL. Se implementará tecnología Hibernate como herramienta de Mapeo objeto-relacional ORM que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación.

#### **6.3.2.2. Lógica de negocio**

Encapsulan las reglas de negocio para procesar los datos que ingresan por la capa de presentación de acuerdo a los requerimientos de negocio. Contemplan varios procesos o pasos dependientes a ejecutarse en un orden determinado. Interactúa con la capa de persistencia para el uso de datos de acuerdo a la etapa del proceso. Para la gestión de reglas de negocio se utilizará la tecnología EJB 3.0

Grafica 6.3.2.2.1. Tecnología EJB 3.0<sup>31</sup>



Fuente: <http://www.jtech.ua.es/j2ee/2003-2004/abierto-j2ee-2003-2004/ejb/sesion01-apuntes.htm>

Los servicios a utilizarse de los EJBs son:

- Manejo de transacciones
- Seguridad y comprobación de permisos de acceso
- Concurrencia
- Servicios de red
- Gestión de recursos
- Persistencia
- Gestión de mensajes
- Escalabilidad
- Adaptación en tiempo de despliegue

### 6.3.2.3. Capa de presentación

Constituye las interfaces gráficas de usuario (GUI) desplegadas en el cliente y los eventos que se ejecutan en el servidor para interactuar a través de un diálogo en línea y proveer de los servicios a los requerimientos del usuario.

Según Kenneth y Julie Kendall un diálogo bien diseñado facilita a las personas el proceso de usar una computadora y reduce su frustración con el sistema de cómputo.

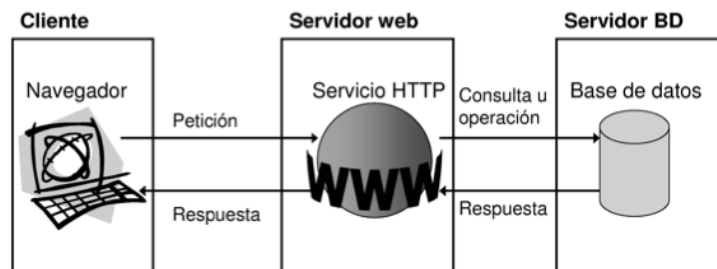
<sup>31</sup> [www.jtech.ua.es/j2ee/2003-2004/abierto-j2ee-2003-2004/ejb/sesion01-apuntes.htm](http://www.jtech.ua.es/j2ee/2003-2004/abierto-j2ee-2003-2004/ejb/sesion01-apuntes.htm)

1. Una comunicación significativa, de manera que la computadora comprenda lo que las personas introducen y las personas comprendan lo que la computadora presenta o solicita.
2. Una mínima acción por parte del usuario.
3. Operación y consistencia estándar en el esquema de pantallas y la distribución de controles.

### 6.3.3. Arquitectura Física

Las capas lógicas serán implementadas en 2 capas físicas. Se destinará un servidor para implementar el motor de base de datos y un servidor para el servidor de aplicaciones Web.

Gráfica 6.3.3.1 Arquitectura física<sup>32</sup>



Fuente: <http://jonaphp.blogspot.com>

#### 6.3.3.1. Servidor de Base de datos

MySQL es un gestor de base de datos orientado a objetos, utilizado en entornos de software libre que no se distancia de muchos servidores de bases de datos relacionales comerciales. Puede funcionar en múltiples plataformas y dispone de varias prestaciones como:

<sup>32</sup> <http://jonaphp.blogspot.com>

- Api de acceso al SGBD, disponible para varios lenguajes de programación
- Varias opciones de conectividad como TCP/IP, socket, ODBC.
- Implementación de algunas extensiones de orientación a objetos.
- Entre otras.

#### **6.3.3.2. Servidor de Aplicaciones Web**

El servidor de aplicaciones Open Source más popular es el JBoss, el cual embebe un servidor de aplicaciones J2EE y otros servicios. Uno de las mayores fortalezas de este servidor es su escalabilidad permitiéndose montar en clústeres ampliables dinámicamente para atender a un crecimiento de la demanda.

### **6.4 Servicios de Subsistemas**

No existen subsistemas, pues la arquitectura que se plantea contempla la capacidad de albergar módulos o subsistemas que interactúen con los ya mencionados como un rastreo satelital de las unidades o interfaces con otros sistemas como el contable a través de servicios web.

En resumen:

La arquitectura distribuida permitirá el acceso a la herramienta de operaciones desde distintos navegadores a través de internet. Esto establece una amplia disponibilidad de trabajo desde el punto de vista geográfico. Además centraliza el mantenimiento y el control del sistema a un solo punto igualmente accesible en la nube.

La división en capas lógicas permite visualizar la asignación de funcionalidades específicas al sistema establecidas en distintos servidores. Las capas lógicas se distribuyen las tareas de gestión y almacenamiento de datos, las reglas de negocio y flujo de datos, y la capa de presentación e interacción con el usuario del sistema.

La fortaleza de la infraestructura presentada es la escalabilidad del sistema de acuerdo a las proyecciones de la empresa. Su plataforma sobre herramientas libres no limitan la implementación de nuevos módulos, mejoras u optimizaciones por aspectos de licencias o soporte técnico.

La solución propuesta se desarrollará en un sistema de código abierto basada en una plataforma madura como es JEE y JBoos Ret Hat, y la base de datos relaciona de libre acceso MySQL. Esto reducirá los costos sin condicionar la robustez de la herramienta, escalabilidad, seguridad en el flujo de información y disposición de información técnica para el desarrollo del sistema. A continuación la tabla comparativa entre las tres plataformas consideradas; de las cuales JAVA fue seleccionado por su orientación a objetos y por su solidez en sistemas de gran escala, portabilidad, seguridad y disponibilidad de componentes de terceros gratuitos.

Por compatibilidad en estructura con JAVA y siendo una base de datos de software libre, se eligió a MySql. Frente a SQL Server y Oracle es un motor más liviano y no requiere de licencias costosas, pero a la vez rápido y seguro. Sin embargo en comparación con PostgreSQL es más robusto y con mayor soporte.



Tabla N.14 Matriz de plataformas

Plataformas	Curva de aprendizaje	Velocidad de desarrollo	IDE (Entorno de desarrollo)	Soporte orientado a objetos	Seguridad	Portabilidad	Rendimiento	Servidor Web	Librerías y frameworks	Coste
J2EE	El más complicado de aprender	Más lento	Herramientas comerciales, Eclipse, Netbeans, otros	excelente	J2EE ofrece una mejor solución para grandes sistemas que corren mediante aplicaciones críticas y múltiples plataformas remotas	multiplataforma	Más pesado, parecido a ASP.NET	Tiene versiones comerciales y open source	la mayoría gratuitas y open source	Puede desarrollarse con herramientas gratuitas y de pago. Por otro punto de vista si se toman las funcionalidades por separado java es más barato.
ASP.NET	Bastante sencillo	El más rápido	Visual Studio (robusta aplicación, pero de coste elevado)	excelente	.NET ofrece mejor respuesta para gestionar autorizaciones basadas en usuarios y roles.	Windows	Más pesado, parecido a J2EE	IIS	altos costos	Licencias bastante caras. Por otro punto de vista si todas las funcionalidades incluidas .Net es más barato que J2EE
PHP	El más sencillo	Rápido	Eclipse, Netbeans, otros	bueno	Intrínsecamente inseguro	multiplataforma	Óptimo en velocidad y uso de recursos	Tiene versiones comerciales y open source	la mayoría gratuitas y open source	Alternativa totalmente gratuita

Elaboración: Johan Cedeño. 2015

Tabla N.15 Características de programa

	BASES DE DATOS			
Características	SQL SERVER	MYSQL	POSTGRESQL	ORACLE
Software libre	NO	SI	SI	NO
Multiplataforma	NO	SI	SI	SI
Soporta multiples lenguajes de programación	BAJO	SI	SI	SI
Cumple propiedades ACID	SI	SI	SI	SI
Mayor velocidad	BAJO	SI	BAJO	SI
Mayor seguridad	BAJO	SI	SI	SI
Soporta SSL	SI	SI	SI	SI

Fuente: <http://dspace.ups.edu.ec/bitstream/>

## Capítulo VII

### 7. Diagramas Estructurales UML

#### 7.1 Introducción

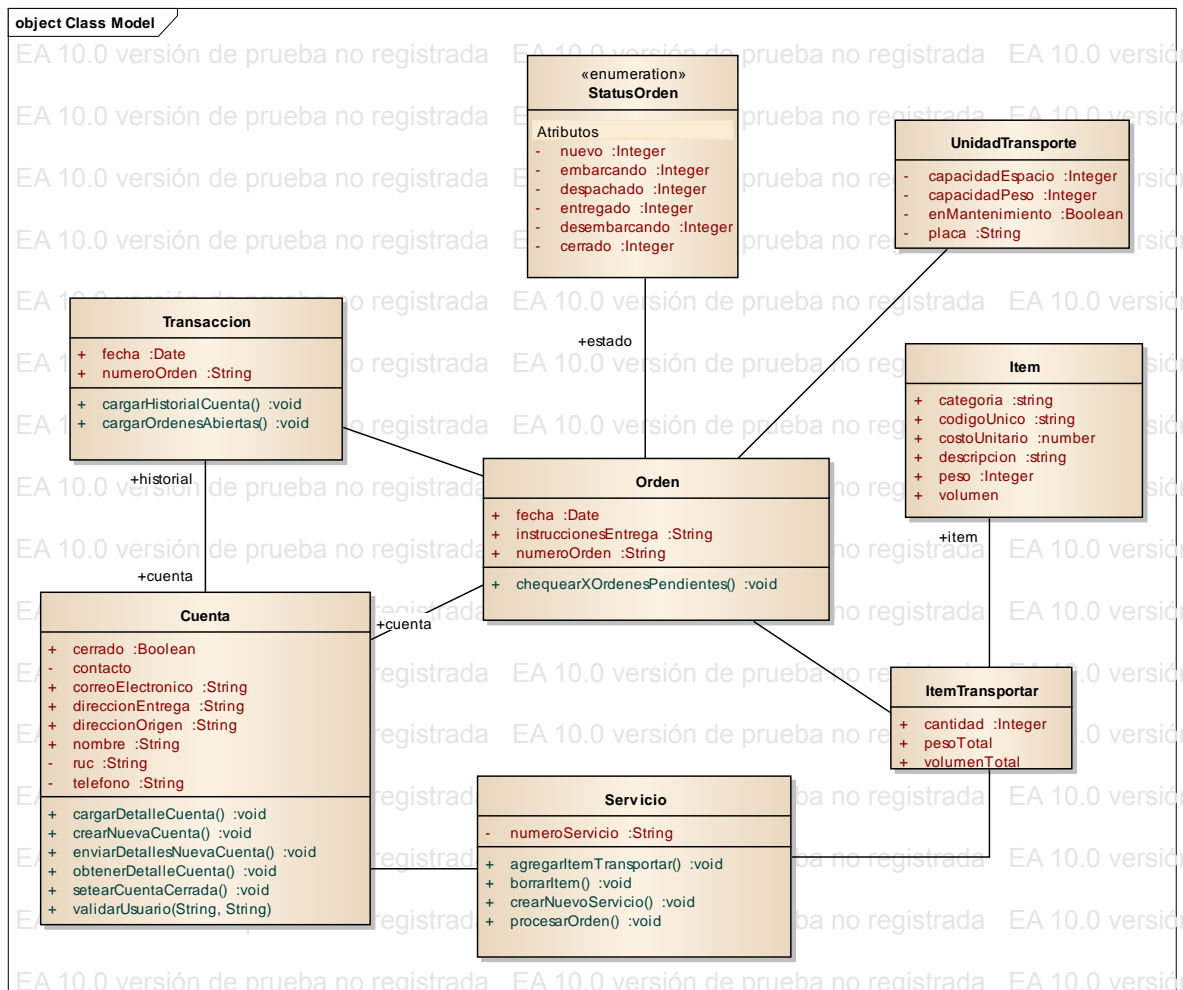
En este capítulo se describirán las entidades básicas del negocio de servicios de transporte de JARCARGO, los actores, unidades de transporte, ítems de mercadería y transacciones que interactúan a través del sistema informático.

#### 7.2 Diagramas de clases

El diagrama de clases muestra las entidades:

- Orden.- Representa la entidad maestro de la unidad de datos que agrupa el pedido de una orden de servicio en una fecha a un cliente en particular.
- ItemTransportar.- Es el detalle de una orden de servicio que asocia un ítem de mercadería con su cantidad, peso y volumen a ser transportada.
- StatusOrden.- Es una colección de datos estáticos que parametrizan los estados que puede caracterizar a un estado.
- Transacción.- Es una tabla histórica que contiene el estado de una orden a una instancia de tiempo específico.
- Cuenta.- Representa un cliente y sus datos.
- Servicio.- Es una clase gestora que se ocupa de las acciones que tienen un pedido.
- Item.- Es la entidad que representa un tipo de mercadería a ser transportada.
- UnidadTrasporte.- Es el automotor que movilizará la mercadería agrupada en una orden, a una cuenta de cliente en una fecha específica.

Gráfica 7.2.1 Diagrama de clases



Elaboración: Johan Cedeño. 2015

Una Orden de transporte de mercadería pasa por diferentes estados desde que se toma el proceso hasta que es cerrada luego del cobro de la factura emitida por el servicio. El resto de clases encapsula los diferentes atributos relacionados a los detalles de la mercadería, el usuario que solicita el servicio o el registro de este evento en la bitácora de transacciones.

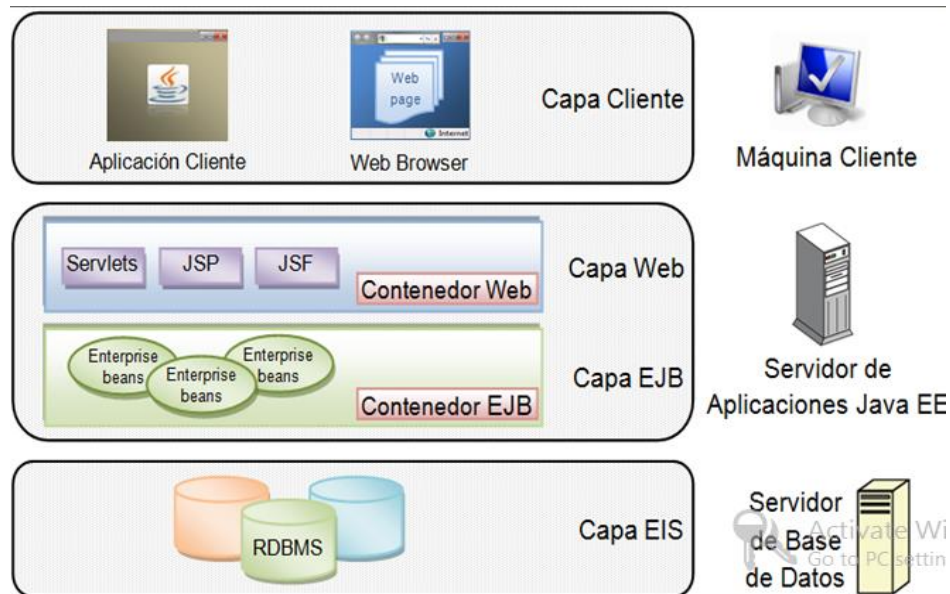
### 7.3 Diagrama de Componentes

Los componentes de la plataforma se distribuyen en tres capas lógicas: persistencia, negocio y presentación; desplegados sobre dos servidores. La capa web con las páginas JSP y la capa

EJBs se instalarán en un solo servidor en plataforma J2EE con JBoss como contener web. El motor de base de datos MySQL se instalará en otro servidor.

La capa de cliente se compone de los scripts que correrán en los navegadores de los usuarios en las distintas sucursales.

Gráfica 7.3.1 Diagrama de componentes



Elaboración: Johan Cedeño. 2015

## 7.4 Diagrama de paquetes

A través de los paquetes se pretende juntar los modelos que representan el sistema complejo que se pretende diseñar. Estos paquetes se han agrupado en dos componentes a nivel de la capa de negocio:

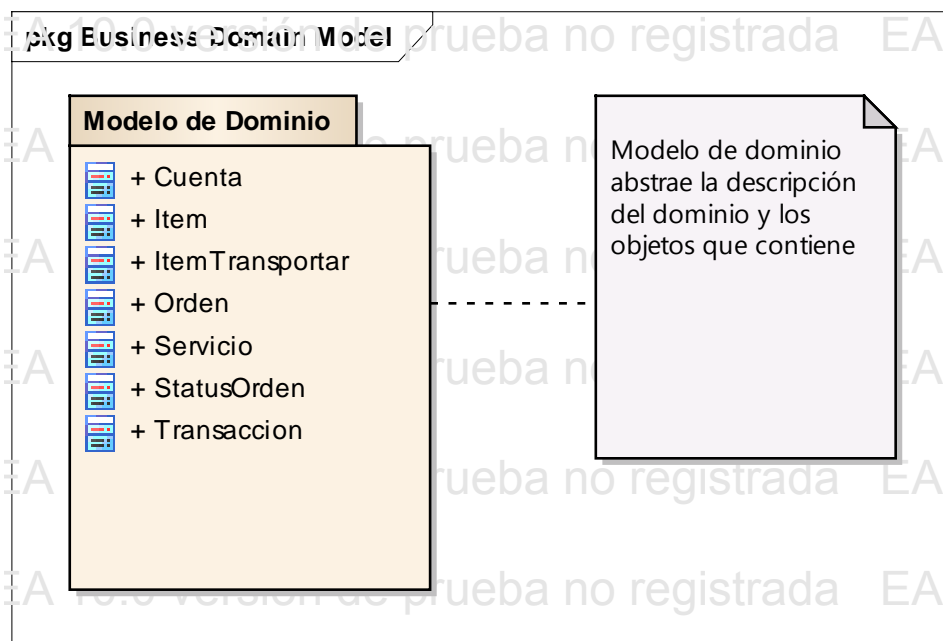
- Los modelos de dominio de negocio
- Los modelos EJBs

### 7.4.1 Modelos de Dominio de Negocio

Los modelos de dominio de negocio encapsulan conceptos como transacciones y ordenes de servicio, para ser gestionados en la capa de negocio en los procesos de creación de órdenes de servicio, seguimiento, movilización, entrega de mercadería y facturación. Dentro de ellos tenemos los Modelos de Dominio y los Modelos de Procesos.

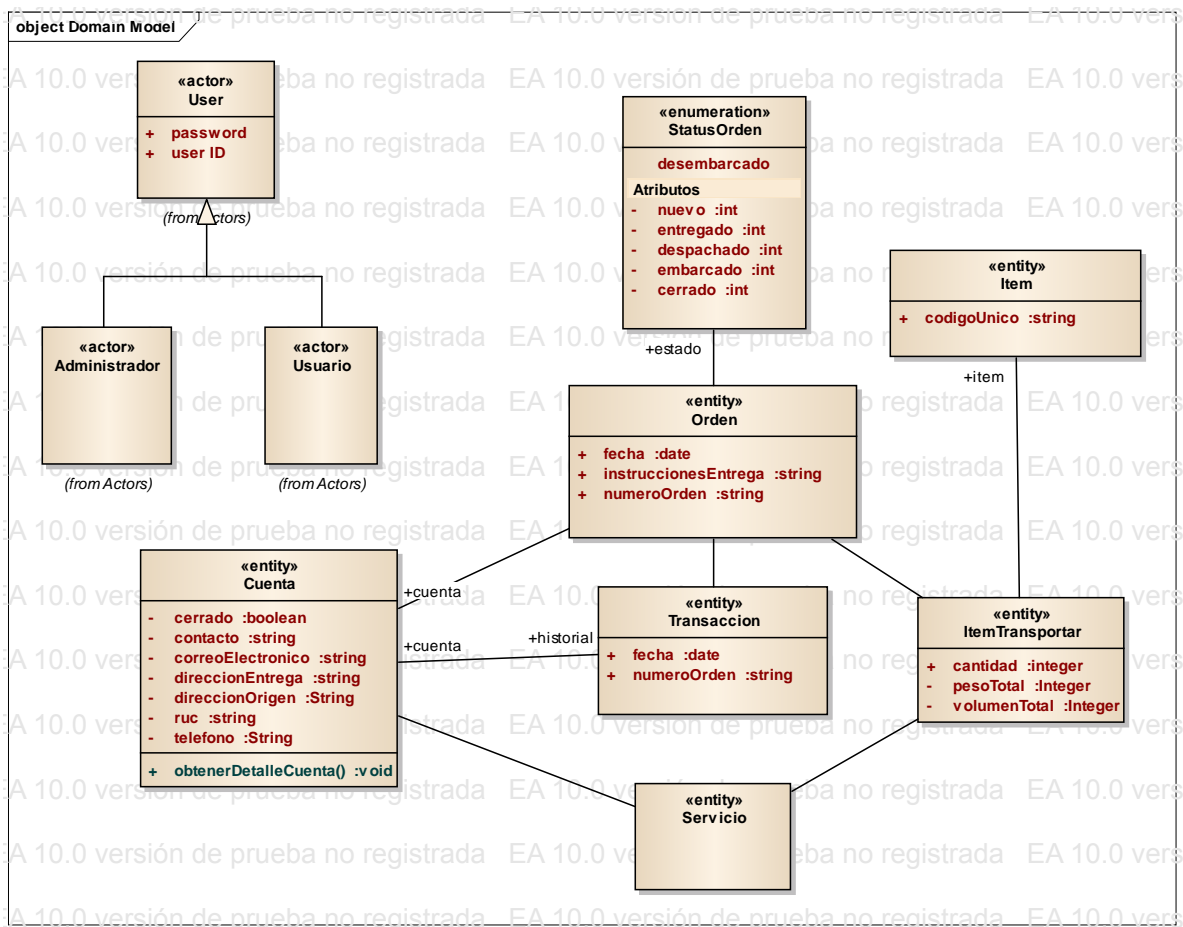
Los Modelos de Dominio son la representación de las clases de la capa de persistencia pero ahora en la capa de negocio, sin embargo agrupan métodos más sofisticados acorde a las necesidades del sistema.

Gráfica 7.4.1.1. Modelos de dominio



Elaboración: Johan Cedeño. 2015

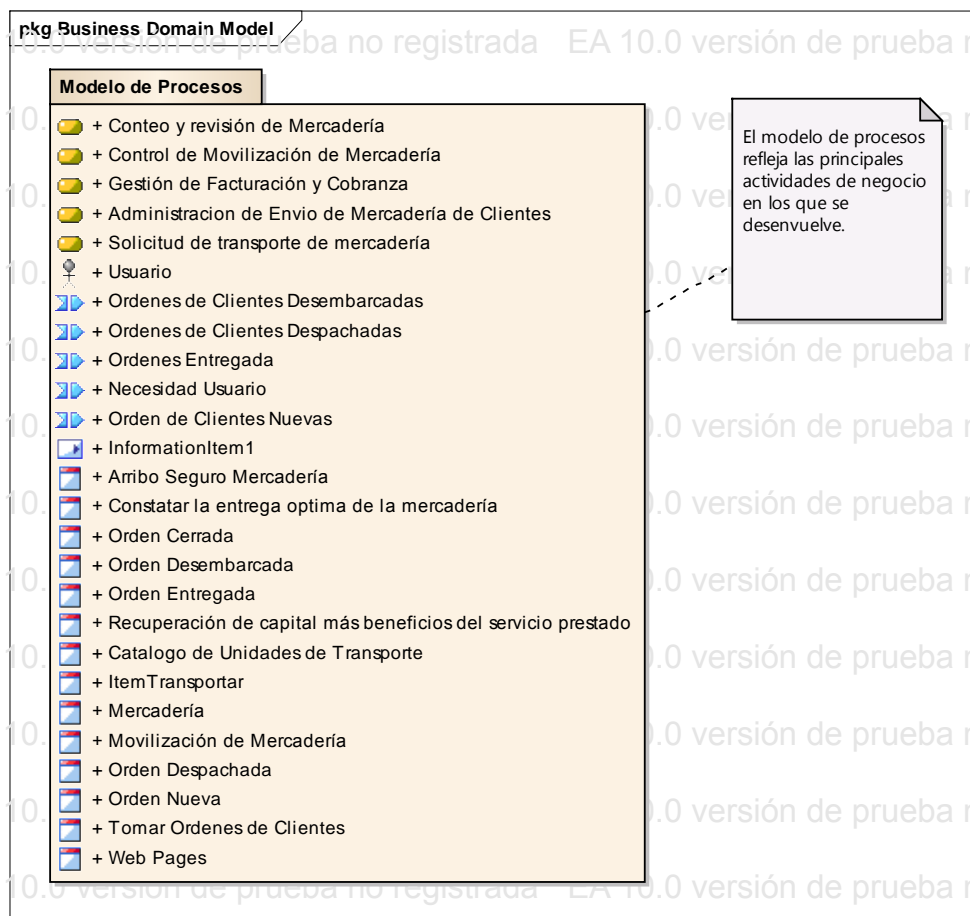
Gráfica 7.4.1.2 Modelos de dominio



Elaboración: Johan Cedeño, 2015

Los modelos de procesos describen los procesos que soportan los requerimientos funcionales levantados y descritos en los ítems 5.3.1 y 5.4.2.

Gráfica 7.4.1.3 Modelos de procesos



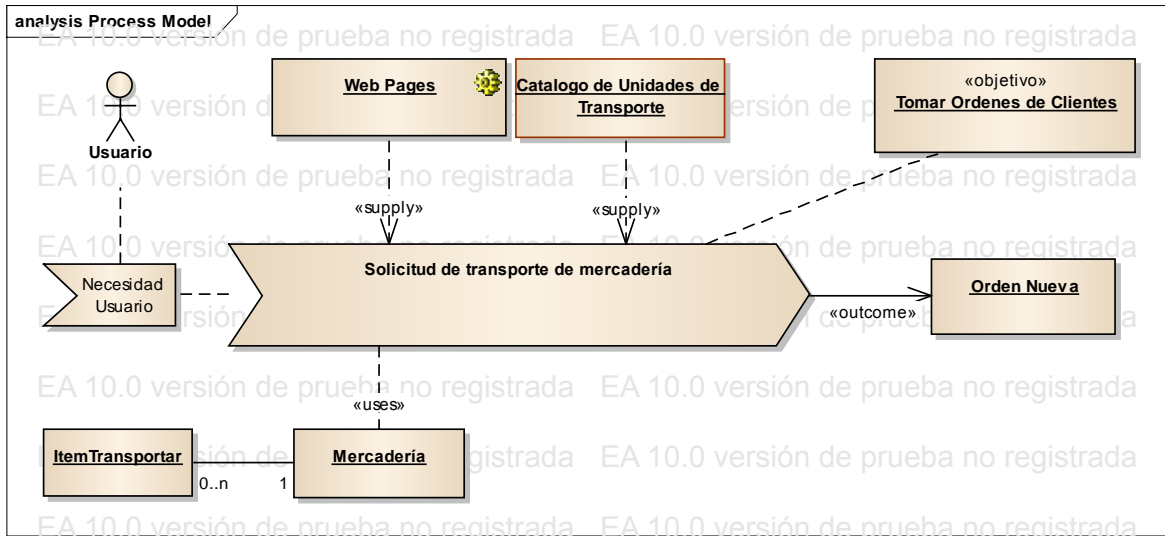
Elaboración: Johan Cedeño. 2015

**Proceso de solicitud de transporte de mercadería.-** Describe el proceso desde que se recibe la petición del cliente vía telefónica o correo para ser ingresada en el sistema.

- Entrada: La necesidad del usuario
- Salida: Creación de una nueva orden de servicio.



Grafica 7.4.1.4 Proceso de solicitud de entrega de mercadería



Elaboración: Johan Cedeño. 2015

**Proceso de administración de envío de mercadería de Clientes.-** Describe el proceso de carga de mercadería en la unidad de transporte asignada en la fecha indicada e inicio de movilización.

- Entrada: Una nueva orden de servicio
- Salida: Despacho de mercadería de una orden en una unidad de transporte.

Gráfica 7.4.1.5. Proceso de envío de mercadería



Elaboración: Johan Cedeño. 2015

**Proceso de control de Movilización de Mercadería.-** Describe el proceso de seguimiento de la unidad desde el punto de partida hasta su destino final.

- Entrada: Una orden de servicio despachada
- Salida: Registro de orden de servicio desembarcada en la bodega o punto solicitado por el cliente.

Gráfica 7.4.1.6. Control de movilización de mercadería

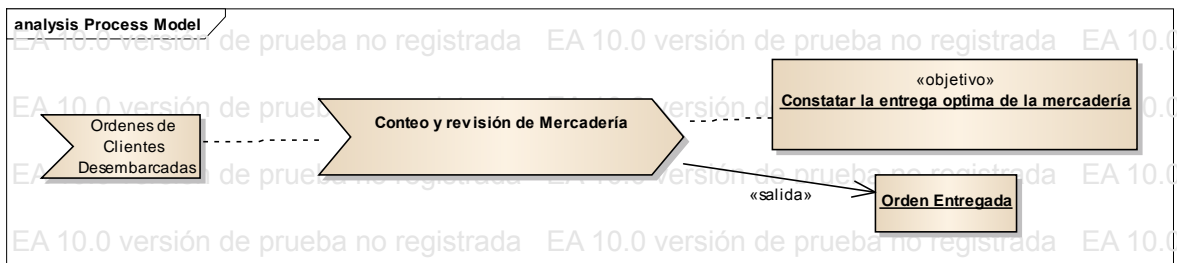


Elaboración: Johan Cedeño. 2015

**Proceso de conteo y Revisión de Mercadería.-** Describe el proceso, conteo y verificación de la mercadería trasladada en las condiciones tales como en el embarque.

- Entrada: Una orden de servicio desembarcada
- Salida: Registro de orden de servicio entregada a conformidad.

Gráfica 7.4.1.7. Proceso de conteo y revisión de mercadería

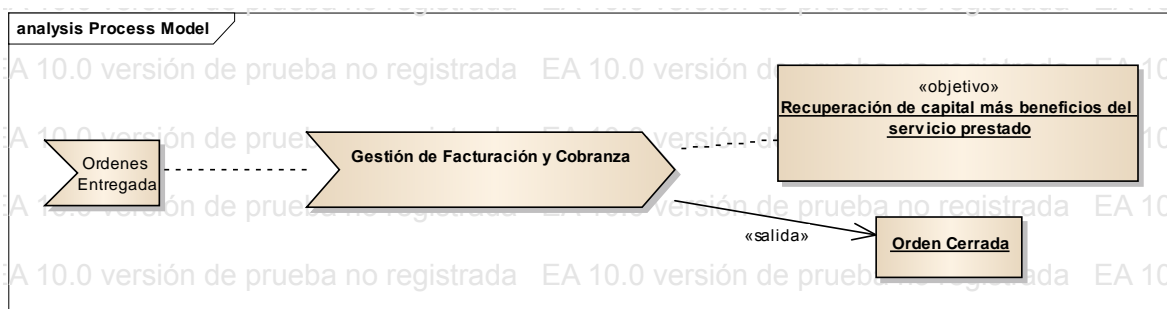


Elaboración: Johan Cedeño. 2015

**Proceso de facturación y Cobranzas.-** Describe el proceso de emisión de facturas a los clientes y la gestión de recuperación de valores por servicio.

- Entrada: Una orden de servicio entregada.
- Salida: Registro de orden de servicio cancelada y cerrada.

Gráfica 7.4.1.8. Proceso de facturación y cobranas



Elaboración: Johan Cedeño. 2015

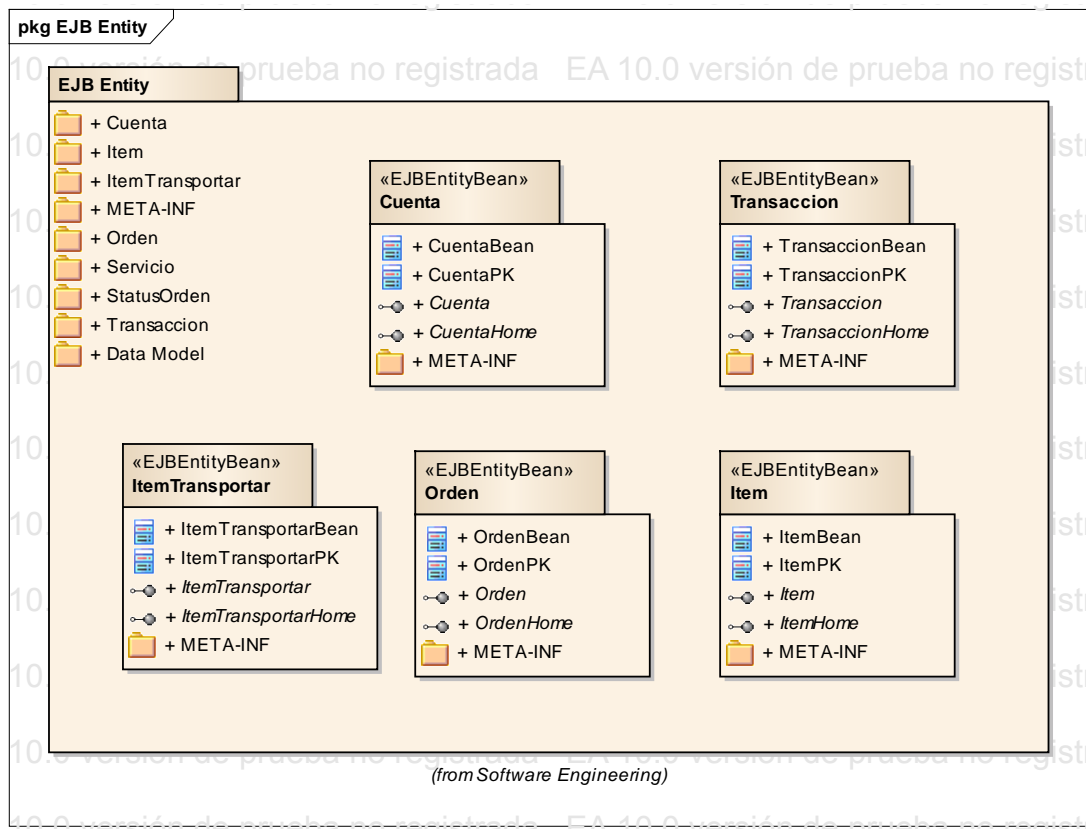
## 7.4.2 Entidades EJBs

Por cada modelo de dominio se creará una entidad EJB para la gestión en la capa de negocio.

De esta manera se tiene:

- Cuenta
- Transacción
- ItemTransportar
- Orden
- Item

Gráfica 7.4.2.1. Entidad EJB



Elaboración: Johan Cedeño. 2015

En resumen:

Las entidades ItemTransportar, StatusOrden, Transacción, Cuenta, Servicio, Item y UnidadTrasporte giran en torno a la clase Orden, que es la entidad que es ocupada por todos los flujos de procesos del sistema.

Hybernate será el interlocutor entre la capa de persistencia y la capa lógica, de tal manera que se aislen lógicamente las dos capas. En la tabla N.16 se muestra que EJB pudo haber sido tomado en cuenta para servir de interlocutor con la capa de persistencia, pero se seleccionó Hybernate porque soporta el mapeo de relaciones complejas de objetos, consultas complejas a la base de datos y obedece a una estandarización de la industria.

Los EJB (de sesión) que encapsulan las funcionalidades con las entidades de dominio serán accedidos a través de las interfaces locales puesto que los módulos correrán sobre la misma máquina virtual.

Tabla N.16 Matriz de herramientas del sistema

	<b>Estándar</b>	<b>Portabilidad</b>	<b>Curva de aprendizaje</b>	<b>Aceptación de la industria</b>	<b>Manejo de seguridad</b>
Hibernate	Hibernate 3.2 es una implementación del estándar.	Se depende de que JBOSS es el único proveedor y se necesita siempre llevar los jars <sup>14</sup> de Hibernate en los proyectos ó en el classpath <sup>15</sup> para que nuestra aplicación funcione cuando se trabaja en otro servidor de aplicaciones.	Muy corta, fácil de aprender. Aunque realmente son pocos los profesionales que realmente usan Hibernate en todo su potencial.	Completamente aceptado por la industria. Existe un gran número de desarrolladores JAVA en el mundo entero que apoyan a Hibernate.	Soportada mediante extensiones de Hibernate y mediante el uso de otros frameworks como Spring.
EJB3	Cada proveedor debe proporcionar una implementación.	Cualquier desarrollo con EJB 3 debe ejecutarse en un servidor de aplicaciones certificado para EJB3 sin realizar cambios en la misma.	En EJB3 es similar a Hibernate. Se debe resaltar que EJB3 tiene un ámbito mucho más amplio que sólo la capa de persistencia. En versiones anteriores de EJB la comparación favorecía a Hibernate.	Existe un fuerte movimiento de los grandes competidores de la industria para adaptar EJB3. Se pueden mencionar a SUN, JBOSS, Oracle, IBM como los más fuertes.	Soportada nativamente como parte de la especificación.

Elaboración: Johan Cedeño. 2015

## Capítulo VIII

### 8.1 Conclusiones

- UML es el lenguaje de modelado más difundido para abstraer un sistema informático. Permite documentar la estructura lógica de los componentes, su interacción y la manera de ser implementada. En la etapa de diseño de un proyecto es una herramienta eficaz para describir todos los elementos actores y procesos a ser implementados en el sistema y que obtienen del levantamiento de requerimientos. En la etapa de desarrollo sirve de mapa suficiente y permanente para la generación de código en base a un diseño original. En la etapa de servicio o soporte permite entender fácilmente la estructura del sistema y brindar una solución efectiva a una modificación del sistema original.
- EL mapa de procesos y el flujograma de operaciones de JARCARGO indican que el sistema procesará de manera lineal y secuencial los pedidos desde que se toman las necesidades de los clientes y la entrega final de la mercadería. Se describen 3 tipos de usuarios del sistema que describen el ámbito de acción: Administración del sistema, Operaciones y Cobranzas.
- Los usuarios del sistema pueden acceder al sistema desde cualquier establecimiento de la empresa a través de conexión a internet. El sistema basado en una plataforma subida a la nube da una amplia cobertura y disponibilidad de los servicios en puntos remotos y dispositivos u ordenadores clientes.
- La herramienta Enterprise Architect, escogida como modelador de UML, dispone de una amplia gama de aplicaciones desde el diseño de los componentes, generador de código, simulador y generador de pruebas del sistema. Si en el camino aún se desea disponer de una herramienta gratuita con amplias prestaciones como esta se recomienda fuertemente Visual Paradigm UML.
- Se organiza lógicamente el sistema por tres capas: Datos, Negocio y Aplicación implantados en 2 sitios servidores. Esta distribución basada en una arquitectura de Java empresarial ampliamente conocida en la comunidad informática de código

abierto. La implementación a través del modelador Enterprise Architect facilita la codificación de reglas de negocio en tecnología EJB 3.0. Las prestaciones esperadas de este sistema son portabilidad, seguridad y disponibilidad de componentes de terceros gratuitos.

- En el modelo de clases, Orden es la entidad que es ocupada por todos los flujos de procesos del sistema para JARCARGO a través de un cambio de estados. Esta al igual que las otras clases básicas serán mapeadas por Hybernate quien será el interlocutor con la capa de Negocio. Los EJB (de sesión) encapsulan las funcionalidades que darán soporte a los procesos de operaciones, cobranzas y mantenimiento.

## 8.2 Recomendaciones

- La documentación creada en UML contempla algunos modelos, como se describe en el capítulo 4, que dependiendo del proyecto y de la minuciosidad de la documentación se puede como no desarrollarlos. En caso de ser necesario en el proyecto actual se debe agregar más diagramas para ampliar su comprensión. Sin embargo, se desarrollará el diseño a partir de los casos de uso que se ajustan a los requerimientos del cliente.
- La administración del desarrollo del proyecto deberá destinar actividades de actualización del modelo UML sobre el que se diseñó el sistema inicialmente. El proceso espiral en el que se basa la implementación contempla posibles cambios de los requerimientos a detalle del cliente y que irán apareciendo en el camino. Esta actualización servirá en la documentación de respaldo para el soporte y mantenimiento del sistema.
- Se debe analizar los diagramas de secuencias del proceso de diseño para describir todos los escenarios posibles que debe contemplar las reglas de negocio que pasan por los Ejbs. En el caso de relevarse nueva información o funcionalidades necesarias estos diagramas deberán ser actualizados y verificados con el contexto del resto de



diagramas.

- Para pasar del levantamiento de requerimientos al análisis y diseño del sistema informático se debe realizar reuniones de trabajo con usuarios de operaciones y contabilidad para validar que la solución satisfaga las funcionalidades de navegabilidad de pantallas, validaciones, flujo de información entre los procesos de ingreso de pedido, gestión de solicitud, transporte y entrega de mercadería, cobranza y cierre de pedido. Con esto se harán las modificaciones necesarias previas a la planificación de trabajo.
- Es importante considerar el enlace de esta herramienta operativa con el sistema contable que dispone Jacargo. El proceso de ingreso de pedidos requiere la correspondiente facturación y su seguimiento con las cuentas por cobrar. Además debe existir la compartición de la base de datos de la información de clientes y de las unidades móviles disponibles, para no tener que duplicar procesos y fuentes de información que al final traen costos para la empresa y riesgos de desactualización de la información.
- Con un diseño completo de UML en la herramienta Architect Enterprise, se puede utilizar la funcionalidad de generador de código que arrojará una estructura de código basada en los modelos creados. Esta estructura es un buen inicio de desarrollo del sistema. Sin embargo no se está restringido a operar de esta manera. Hay desarrolladores que prefieren iniciar la codificación desde cero, para un mayor control de las versiones de código.

## **Bibliografía**

Adingor. (2014). *www.adingor.es*.

AEADE. (2014). *www.aeade.net*. Obtenido de

[http://www.aeade.net/web/index.php?option=com\\_content&view=article&id=145&Itemid=80](http://www.aeade.net/web/index.php?option=com_content&view=article&id=145&Itemid=80)

Bermón, L. (2014).

<http://www.virtual.unal.edu.co/cursos/sedes/manizales/4060010/lecciones/Capitulo1/modelo.htm>. Obtenido de

<http://www.virtual.unal.edu.co/cursos/sedes/manizales/4060010/lecciones/Capitulo1/modelo.htm>

FENATRAPE. (2014). <http://www.fenatrape.org/index.php>. Obtenido de

<http://www.fenatrape.org/index.php>

<http://lapastillaroja.net>. (2014). <http://lapastillaroja.net>. Obtenido de

<http://lapastillaroja.net/2013/07/app-escalable/>

Orallo, E. H. (2001). *C++ estándar*. Editorial Paraninfo.

Perez, L. (2012). [http://softwareverde.blogspot.com/2012\\_09\\_01\\_archive.html](http://softwareverde.blogspot.com/2012_09_01_archive.html). Obtenido

de [http://softwareverde.blogspot.com/2012\\_09\\_01\\_archive.html](http://softwareverde.blogspot.com/2012_09_01_archive.html)

Wikipedia. (2013). [http://es.wikipedia.org/wiki/Lenguaje\\_unificado\\_de\\_modelado](http://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado).

Obtenido de [http://es.wikipedia.org/wiki/Lenguaje\\_unificado\\_de\\_modelado](http://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado)

[www.unitty.com](http://www.unitty.com). (2014). [www.unitty.com](http://www.unitty.com). Obtenido de

[http://www.unitty.com/index.php?option=com\\_k2&view=item&id=139:ea-](http://www.unitty.com/index.php?option=com_k2&view=item&id=139:ea-)

## ANEXOS

### Glosario de términos

- MDA(del inglés (Model-Driven, Architecture, arquitectura dirigida por modelos es un acercamiento al diseño de software, propuesto y patrocinado por el Object Management Group. MDA se ha concebido para dar soporte a la ingeniería dirigida a modelos de los sistemas de software)
- XMI o XML Metadata Interchange (XML de Intercambio de Metadatos) es una especificación para el Intercambio de Diagramas.

- TOGAF.- The Open Group Architecture Framework o Esquema de Arquitectura de Open Group, en español) es un esquema de Arquitectura Empresarial que proporciona un enfoque para el diseño, planificación, implementación y gobierno de una arquitectura empresarial de información.
- BPMN.- Business Process Model and Notation o Modelo y Notación de Procesos de Negocio, es una notación gráfica estandarizada que permite el modelado de procesos de negocio, en un formato de flujo de trabajo.
- WSDL.- Web Services Description Language, un formato XML que se utiliza para describir servicios Web.
- JBoss.- Es un servidor de aplicaciones Java EE de código abierto implementado en Java puro de propiedad de Red Hat desde abril del 2006. JBoss AS es el primer servidor de aplicaciones de código abierto, preparado para la producción y certificado J2EE 1.4.
- Hybernate.- es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación.
- AEADE.- Asociación de empresas automotrices del Ecuador
- FENATRAPE.- Federación Nacional de Transporte pesado del Ecuador